대 한 민 국 특 허 청
KOREAN INTELLECTUAL PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출 원 번 호 : 10-2003-0011133
Application Number

출 원 년 월 일 : 2003년 02월 21일
Date of Application FEB 21, 2003

출 원 인 : 삼성전자주식회사
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.

2003 년 06 월 03 일

특 허 청
COMMISSIONER

【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0003
【제출일자】	2003.02.21
【국제특허분류】	H04B
【발명의 명칭】	이동통신시스템에서 인터넷 프로토콜 버전에 따른 트래픽 플로우 탬플릿 패킷 필터링 장치 및 방법
【발명의 영문명칭】	APPARATUS FOR TRAFFIC FLOW TEMPLATE PACKET FILTERING ACCORDING TO INTERNET PROTOCOL VERSION IN MOBILE COMMUNICATION SYSTEM AND METHOD THEREOF
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이건주
【대리인코드】	9-1998-000339-8
【포괄위임등록번호】	2003-001449-1
【발명자】	
【성명의 국문표기】	안홍진
【성명의 영문표기】	AHN,Hong-Jin
【주민등록번호】	730306-1279626
【우편번호】	442-813
【주소】	경기도 수원시 팔달구 영통동 1006-8 105호
【국적】	KR
【취지】	특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인 이건주 (인)
【수수료】	
【기본출원료】	20 면 29,000 원
【가산출원료】	69 면 69,000 원
【우선권주장료】	0 건 0 원
【심사청구료】	0 항 0 원
【합계】	98,000 원

【요약서】

【요약】

본 발명은 제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서, 소스 IP 어드레스의 버전에 상응하여 상기 소스 IP 어드레스로부터 상기 IP 버전에 따른 정보를 추출하는 과정과, 상기 추출한 정보를 포함하는 TFT 정보를 생성하여 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 과정을 포함한다.

【대표도】

도 18a

【색인어】

TFT, IPv4 embedded IPv6 어드레스, IPv4 compatible IPv6 어드레스, IPv4 mapped IPv6 어드레스

【명세서】

【발명의 명칭】

이동통신시스템에서 인터넷 프로토콜 버전에 따른 트래픽 플로우 탬플릿 패킷 필터링 장치 및 방법{APPARATUS FOR TRAFFIC FLOW TEMPLATE PACKET FILTERING ACCORDING TO INTERNET PROTOCOL VERSION IN MOBILE COMMUNICATION SYSTEM AND METHOD THEREOF}

【도면의 간단한 설명】

도 1은 일반적인 UMTS 네트워크 구조를 개략적으로 도시한 도면

도 2는 일반적인 TFT가 사용되는 UMTS 코어 네트워크를 개략적으로 도시한 도면

도 3은 일반적인 TFT 구조를 도시한 도면

도 4는 제1 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 도시한 신호 흐름도

도 5는 제2 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 도시한 신호 흐름도

도 6은 새로운 TFT 생성을 위한 TFT 정보들을 개략적으로 도시한 도면

도 7은 일반적인 IPv6 어드레스 구조를 개략적으로 도시한 도면

도 8은 저장되어 있는 TFT를 삭제하거나 저장되어 있는 TFT에 패킷 필터를 첨가하거나 혹은 패킷 필터 대치를 하기 위한 TFT 정보들을 개략적으로 도시한 도면

도 9는 저장되어 있는 TFT 패킷 필터를 삭제하기 위한 TFT 정보들을 개략적으로 도시한 도면

도 10은 일반적인 UMTS 코어 네트워크에서 TFT 패킷 필터링 과정을 개략적으로 도시한 도면

도 11은 일반적인 IPv4 compatible IPv6 어드레스 구조를 개략적으로 도시한 도면

도 12는 IPv4 compatible IPv6 어드레스가 사용되는 네트워크 구조를 개략적으로 도시한 도면

도 13은 일반적인 IPv4 mapped IPv6 어드레스 구조를 개략적으로 도시한 도면

도 14는 IPv4 mapped IPv6 어드레스가 사용되는 네트워크 구조를 개략적으로 도시한 도면

도 15는 본 발명의 실시예에서의 기능을 수행하기 위한 UMTS 네트워크 구조를 개략적으로 도시한 도면

도 16은 본 발명의 실시예에서의 기능을 수행하기 위한 TFT 패킷 필터링 장치 내부 구조를 도시한 블록도

도 17은 도 16의 TFT 테이블(1651)에 저장되는 TFT 정보를 도시한 도면

도 18a 내지 도 18b는 IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링 과정을 도시한 순서도

도 19a 내지 도 19b는 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링 과정을 도시한 순서도

도 20은 도 16의 TFT 패킷 필터링 프로시저(1611)의 일반적인 TFT 패킷 필터링 동작을 개략적으로 도시한 도면

도 21은 도 16의 TFT 패킷 필터링 프로시저(1611)가 IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 동작을 개략적으로 도시한 도면

도 22는 도 16의 TFT 패킷 필터링 프로시저(1611)가 IPv4 Embedded IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 동작을 개략적으로 도시한 도면

도 23은 본 발명의 IPv6 source address type 방법 및 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링에 따른 비트 연산량과 일반적인 TFT 패킷 필터링에 따른 비트 연산량을 비교적으로 도시한 도면

도 24는 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터 생성 과정을 도시한 순서도

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<25> 본 발명은 이동 통신 시스템에 관한 것으로서, 특히 인터넷 프로토콜 어드레스 버전에 따라 트래픽 플로우 템플릿 패킷 필터링을 수행하는 장치 및 방법에 관한 것이다.

<26> 이동통신시스템(Mobile Communication System)인 UMTS(Universal Mobile Telecommunications Systems, 이하 "UMTS"라 칭하기로 한다)은 제3세대(3rd Generation) 이동 통신을 수행하는 시스템이다. 상기 UMTS 시스템은 음성 서비스뿐만 아니라 패킷 데이터(packet data) 서비스를 지원하고, 고속 데이터 통신 및 동영상 통신 등을 지원한다. 상기 UMTS 네트워크(network)의 개략적인 구조를 도 1을 참조하여 설명하기로 한다.

<27> 상기 도 1은 일반적인 UMTS 네트워크 구조를 개략적으로 도시한 도면이다.

<28> 상기 도 1을 참조하면, 먼저 사용자 단말기(UE: User Equipment, 이하 "UE"라 칭하기로 한다)(111)는 UMTS 육상 무선 접속 네트워크(UTRAN: UMTS Terrestrial Radio Access Network, 이하 "UTRAN"이라 칭하기로 한다)(113)와 접속되어 호(call)를 처리하며, 회선 서비스(CS: Circuit Service)와 패킷 서비스(PS: Packet Service)를 모두 지원한다. 상기 UTRAN(113)은 기지국(Node B)(도시하지 않음)과, 무선 네트워크 제어기(RNC: Radio Network Controller, 이하 "RNC"라 칭하기로 한다)(도시하지 않음)로 구성되며, 상기 기지국은 상기 UE(111)과 Uu 인터페이스(interface)를 통해서 연결되며, 상기 RNC는 서비스 패킷 무선 서비스 지원 노드(SGSN: Serving GPRS Support Node, 이하 "SGSN"이라 칭하기로 한다)(115)와 Iu 인터페이스를 통해서 연결된다. 여기서, 상기 패킷 무선 서비스(GPRS: General Packet Radio Service, 이하 "GPRS"라 칭하기로 한다)는 상기 UMTS 네트워크에서 수행하는 패킷 데이터 서비스이다. 상기 UTRAN(113)은 상기 UE(111)에서 에어(air)상으로 전송한 무선 데이터 혹은 제어 메시지(control message)들을 GPRS 터널링 프로토콜(GTP: GPRS Tunneling Protocol, 이하 "GTP"라 칭하기로 한다)을 사용하는 코어 네트워크(CN: Core Network)로 전달하기 위해 프로토콜 변환을 수행한다. 여기서, 상기 코어 네트워크는 상기 SGSN(115)과 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node, 이하 "GGSN"이라 칭하기로 한다)(119)를 통칭한다.

<29> 그리고, 상기 SGSN(115)는 UE(111)의 가입자 정보와, 위치 정보를 관리하는 네트워크 노드이다. 상기 SGSN(115)는 상기 UTRAN(113)과는 Iu 인터페이스를 통해 연결되며, GGSN(119)과는 Gn 인터페이스를 통해 연결되어 데이터 및 제어 메시지 등을 송수신한다. 그리고 상기 SGSN(115)는 홈위치 등록기(HLR: Home Location

Register)(117)와 Gr 인터페이스를 통해 연결되어 상기 가입자 정보 및 위치 정보를 관리한다.

<30> 상기 홈위치 등록기(117)는 패킷 도메인(packet domain)의 가입자 정보 및 라우팅(routing) 정보 등을 저장한다. 상기 홈위치 등록기(117)는 상기 SGSN(115)과는 Gr 인터페이스를 통해 연결되며, 상기 GGSN(119)과는 Gc 인터페이스를 통해 연결된다. 그리고, 상기 홈위치 등록기(117)는 UE(111)의 로밍(roaming)등을 고려하여 다른 공중 육상 이동 통신 네트워크(PLMN: Public Land Mobile Network, 이하 "PLMN"이라 칭하기로 한다)에 위치할 수 있음은 물론이다. 그리고 상기 GGSN(119)은 상기 UMTS 네트워크에 있어서 GTP의 종단이며, Gi 인터페이스를 통해 외부 네트워크와 연결되어 인터넷(internet)(121), 혹은 패킷 도메인 네트워크(PDN: Packet Domain Network), 혹은 다른 PLMN등과 연동할 수 있다.

<31> 다음으로 도 2를 참조하여 트래픽 플로우 탬플릿(TFT: Traffic Flow Template, 이하 "TFT"라 칭하기로 한다)이 사용되는 UMTS 코어 네트워크의 구조를 개략적으로 설명하기로 한다.

<32> 상기 도 2는 일반적인 TFT가 사용되는 UMTS 코어 네트워크를 개략적으로 도시한 도면이다.

<33> 상기 도 2를 설명하기에 앞서 먼저 상기 TFT를 사용한다는 것은 TFT를 사용하여 패킷 필터링(packet filtering)을 수행한다는 것을 의미하며, 상기 TFT 사용은 UMTS 코어 네트워크에서 이루어진다. 상기 TFT 사용을 설명하면 다음과 같다. 먼저 패킷 데이터 프로토콜(PDP: Packet Data Protocol, 이하 "PDP"라 칭하기로 한다) 컨텍스트(context)는 제1 PDP 컨텍스트(primary PDP context)와 제2 PDP 컨텍스트(second PDP context)의 2가

지 종류가 존재한다. 상기 제2 PDP 컨텍스트는 상기 제2 PDP 컨텍스트와 동일한 정보를 가지는 PDP 컨텍스트, 즉 제1 PDP 컨텍스트가 존재할 경우에만 존재하는 것이 가능하다. 즉, 제2 PDP 컨텍스트는 제 1 PDP 컨텍스트의 정보를 그대로 재사용하는 것이기 때문에 상기 제1PDP 컨텍스트가 생성된 이후에야 생성 가능하다. 이렇게 상기 제1 PDP 컨텍스트와 제2 PDP 컨텍스트는 실제 사용하는 정보는 동일하고, 다만 실제 패킷 데이터가 전송되는 GTP 터널만이 상이하다.

<34> 특히 UMTS 코어 네트워크에서는 상기 제 2PDP 컨텍스트를 활성화시킬(activate) 경우, 제1 PDP 컨텍스트와 제 2 PDP 컨텍스트를 구분하기 위한 필터(filter)로서 상기 TFT 정보를 사용하게 된다. 상기 도 2에 도시한 바와 같이 UMTS 코어 네트워크(200), 즉 광대역 부호 분할 다중 접속(WCDMA: Wideband Code Division Multiple Access)(200)에는 7개의 TFT들이 저장되어 있으며, 상기 7개의 TFT들에 상응하는 제2PDP 컨텍스트들과 제1 PDP 컨텍스트를 고려하여 총 8개의 GTP 터널들이 생성되어 있다. 외부 네트워크, 일 예로 인터넷(121)을 통해 유입되는 인터넷 프로토콜(IP: Internet Protocol, 이하 "IP"라 칭하기로 한다) 패킷 데이터는 Gi 인터페이스를 통해서 GGSN(119)로 입력된다. 상기 GGSN(119)에는 7개의 TFT들, 즉 TFT1부터 TFT 7이 저장되어 있으며, 상기 Gi 인터페이스를 통해 입력되는 IP 패킷 데이터에 사용할 패스는 상기 저장되어 있는 7개의 TFT들을 통해 패킷 필터링을 통해 결정된다. 그리고 상기 GGSN(119)에서 TFT를 사용하여 필터링된 IP 패킷 데이터는 결정된 패스, 즉 결정된 GTP 터널로 Gn 인터페이스를 통해서 SGSN(115)로 전달되고, 상기 SGSN(115)는 상기 GGSN(119)로부터 전달받은 IP 패킷 데이터를 해당 GTP 터널로 Iu 인터페이스를 통해 무선 접속 네트워크(RAN: Radio Access Network)(211)로 전달한다.

<35> 다음으로 도 3을 참조하여 상기 TFT 구조를 설명하기로 한다.

<36> 상기 도 3은 일반적인 TFT 구조를 도시한 도면이다.

<37> 먼저, TFT는 UE(111)에서 생성되며, 상기 생성된 TFT는 UTRAN(113) 및 SGSN(115)를 통해 GGSN(119)로 전달된다. 그리고 상기 GGSN(119)는 제1 GTP(Primary GTP) 터널과 제2 GTP(Secondary GTP) 터널들을 구분하기 위해 TFT를 사용하여 외부 네트워크, 일 예로 인터넷(121)을 통해 입력되는 패킷 데이터를 필터링하여 상기 패킷 데이터가 실제 전송될 GTP 터널을 찾게 되는 것이다. 그리고 제1 PDP 컨텍스트를 사용하는 제1 GTP 터널과 제2 PDP 컨텍스트를 사용하는 제2 GTP 터널들은 각각 PDP 어드레스(address)가 동일하므로 실제로 TFT가 존재하지 않을 경우, 외부 네트워크로부터 수신되는 패킷 데이터들이 어떤 GTP 터널을 통해 전송되는지, 즉 제1 GTP 터널을 통해 전송되는지 혹은 제2 GTP 터널을 통해 전송되는지를 구별하는 것이 불가능하게 된다.

<38> 그리고 상기 TFT는 고유한 패킷 필터 ID(packet filter identifier)에 의해 구분되는 패킷 필터를 복수개, 일 예로 8개까지 가질 수 있다. 상기 패킷 필터는 동일한 PDP 어드레스를 공유하는 PDP 컨텍스트와 관련된 모든 TFT들에 대해서 고유한 평가 순위 인덱스(evaluation precedence index)를 가지고 있다. 상기 평가 순위 인덱스는 255에서 0 사이의 값들 중 하나의 값을 가지는데, 상기 UE(111)는 패킷 필터 ID와 패킷 필터의 평가 순위 인덱스를 관리하며, 실제 패킷 필터의 컨텐츠(contents)를 생성한다. 또한, 상기 TFT는 제2 PDP 컨텍스트 활성화 과정에 있어서 항상 PDP 컨텍스트와 일대일 대응된다. 즉, 상기 TFT는 PDP 컨텍스트 활성화 과정에서 생성된 PDP 컨텍스트에 상기 UE(111)에 의한 PDP 컨텍스트 수정 과정(MS-Initiated PDP Context Modification Procedure)을 통해 추가 생성 가능하며, 상기 UE(111)에 의한 PDP 컨텍스트 수정 과정

(MS-Initiated PDP Context Modification Procedure)을 통해 수정도 가능하다. 여기서, 하나의 PDP 컨텍스트는 하나 이상의 TFT를 가질 수 없다.

<39> 상기 도 3을 참조하면, 상기 TFT는 TFT 타입(Traffic Flow Template Type) 영역과, TFT 타입 길이(Length of Traffic Flow Template Type) 영역과, TFT 연산 코드(TFT operation code) 영역과, 패킷 필터 수(number of packet filters) 영역과, 패킷 필터 리스트(Packet filter List) 영역을 가진다. 상기 TFT 타입 영역은 사용되는 TFT의 타입을 나타내는 영역으로서, 일반적으로 UMTS 코어 네트워크에서(200)는 그 값(value)을 137로 설정하며, 네트워크에 따라서 상이하게 설정 가능함은 물론이다. 그리고 상기 TFT 타입 길이 영역은 사용되는 TFT 타입의 길이를 나타내는 영역이며, 소정 길이, 일 예로 2 바이트(Byte)의 영역 크기를 가지며, 상기 TFT 타입 영역과 상기 TFT 타입 길이 영역을 제외한 나머지 영역의 크기를 나타낸다. 그리고, TFT 연산 코드 영역은 사용되는 TFT의 연산 코드를 나타내는 영역이며, 상기 TFT 연산 코드 영역에 나타나 있는 값을 해석해서 UE(111)으로부터 수신한 TFT를 어떤 방식으로 처리할 것인지를 결정하게 된다. 상기 TFT 연산 코드 영역에서 나타내는 코드들을 하기 표 1에 나타내었다.

<40>

【표 1】

Bits (765)	내 용
000	Spare
001	새로운 TFT를 생성한다.
010	저장 중인 TFT를 삭제한다.
011	저장 중인 TFT에 패킷 필터를 첨가한다.
100	저장 중인 TFT의 패킷 필터와 대치한다.
101	저장 중인 TFT의 패킷 필터를 삭제한다.
110	예약되어 있음
111	예약되어 있음

<41> 상기 표 1에 도시한 바와 같이, TFT 연산 코드 "000"은 spare 값을 나타내며, TFT 연산 코드 "001"은 새로운 TFT를 생성하는 연산을 나타내며, TFT 연산 코드 "010"은 저장중인 TFT를 삭제한다는 연산을 나타내며, TFT 연산 코드 "011"은 저장중인 TFT에 패킷 필터를 첨가한다는 연산을 나타내며, TFT 연산 코드 "100"은 저장중인 TFT의 패킷 필터와 대치한다는 연산을 나타내며, TFT 연산 코드 "101"은 저장중인 TFT의 패킷 필터를 삭제한다는 연산을 나타내며, TFT 연산 코드 "110" 및 "111"은 reserved를 나타낸다. 상기 GGSN(119)은 상기 TFT 연산 코드 영역을 읽어 해당 연산을 수행하게 된다.

<42> 그리고 상기 패킷 필터 수 영역은 사용되는 TFT에 설정되어 있는 패킷 필터들의 수를 나타내는 영역으로서, 상기 TFT의 패킷 필터 리스트에 존재하는 패킷 필터들의 수를 나타낸다. 예를 들어 TFT 연산 코드 영역의 값이 "010"으로 저장되어 있는 경우, 즉 저장중인 TFT를 삭제하는 경우 상기 패킷 필터 수 영역의 값은 0으로 설정된다. 그래서 상기 저장중인 TFT를 삭제하는 경우 이외의 상기 패킷 필터수 영역의 값은 0보다는 크고 8 이하가 되도록 설정된다(0 < number of packet filters ≤8). 여기서, 상기 패킷 필터수

영역의 값이 0보다는 크고 8이하가 되도록 설정하는 이유는 상기 UMTS 코어 네트워크(200)에서 사용하는 패킷 필터수를 최대 8개로 설정하였기 때문이다. 그리고 상기 TFT 정보에는 상기 패킷 필터가 최소 1개 최대 8개까지를 가지도록 할 수 있다. 그리고 상기 패킷 필터는 그 컨텐츠가 하나인 단일 필드 패킷 필터(single-field filter)와 그 컨텐츠가 다수개로 구성된 멀티 필드 패킷 필터(multi-field packet filter)로 구분된다. 여기서, 상기 단일 필드 패킷 필터는 패킷 필터에서 필터링하는 컨텐츠가 한 개, 일 예로 소스 어드레스(source address)와 같은 한 개의 컨텐츠로 구성되며, 상기 멀티 필드 패킷 필터는 패킷 필터에서 필터링하는 컨텐츠가 다수개, 일 예로 소스 어드레스와, 프로토콜과, 데스티네이션 어드레스(destination address)와 같은 다수개의 컨텐츠들로 구성된다. 상기 패킷 필터 리스트 영역은 상기 TFT에 설정되는 패킷 필터들의 실제 사용 정보들에 대한 내용을 나타내는 영역이다.

<43> 상기 도 3과 같은 구조를 가지는 TFT가 GGSN(119)에 저장되어 있고, 외부 인터넷(121)으로부터 IP 패킷 데이터가 입력되면 상기 저장되어 있는 TFT 내에 저장되어 있는 패킷 필터들을 통해 필터링된다. 여기서, 상기 TFT 내의 패킷 필터들에 의해 필터링되는 IP 패킷 데이터들은 해당 TFT가 저장된 PDP 컨텍스트를 사용하게 된다. 그래서, 입력되는 IP 패킷 데이터가 TFT 내의 다수의 패킷 필터들중, 일 예로 TFT 내에 제1패킷 필터부터 제3필터까지 3개의 패킷 필터가 존재할 경우 그 3개의 패킷 필터들중 첫 번째 패킷 필터인 제1패킷 필터를 만족하지 않는다면, 상기 TFT에 저장되어 있는 다음 패킷 필터, 즉 두 번째 패킷 필터인 제2패킷 필터를 적용한다. 이런 식으로 마지막 패킷 필터까지 모든 패킷 필터들을 만족하지 않는다면 상기 입력된 IP 패킷 데이터는 다른 GTP 터널을

사용하는 것이며, 상기 패킷 필터링이 종료된 TFT가 아닌 다음 TFT를 사용하여 패킷 필터링을 시도하게 된다.

<44> 다음으로 도 4를 참조하여 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 설명하기로 한다.

<45> 상기 도 4는 제1 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 도시한 신호 흐름도이다.

<46> 먼저, UMTS 패킷 도메인에서 데이터, 즉 패킷 데이터를 전송하기 위해서는 상기 패킷 데이터를 전송하기 위한 GTP 터널을 생성해야만 한다. 상기 GTP 터널이 생성되는 경로는 크게 UE(111)가 코어 네트워크에 요청하는 경우, 즉 UE 초기 활성화(UE-Initiated Activate)와 외부 네트워크에서 상기 UMTS 코어 네트워크에 요청하는 경우, 즉 네트워크 요청 활성화(Network Requested Activate)의 두 가지 경로로 구분된다.

<47> 상기 도 4를 참조하면, UE(111)는 패킷 데이터의 발생을 감지함에 따라 상기 패킷 데이터를 전송하기 위해서 GTP 터널을 생성하게 된다. 이렇게 UE(111)는 GTP 터널 생성을 위해 SGSN(115)으로 PDP 컨텍스트 활성화 요청(Activate PDP Context Request) 메시지(message)를 전송한다(411단계). 상기 PDP 컨텍스트 활성화 요청 메시지에 포함되는 파라미터(parameter)들로는 네트워크 계층 서비스 접속 포인트 식별자(NSAPI: Network layer Service Access Point Identifier, 이하 "NSAPI"라 칭하기로 한다)와, TI와, PDP 타입(type)과, PDP 어드레스(address)와, 접속 포인트 명(Access Point Name)과, 서비스 품질(QoS: Quality of Service)등이 있다.

<48> 여기서, 상기 NSAPI는 상기 UE(111)에서 생성되는 정보로서, 5번에서 15번까지 총 11개의 값을 사용할 수 있다. 상기 NSAPI 값은 PDP 어드레스와, PDP 컨텍스트 ID(PDP Context Identifier)와 일대일 대응된다. 상기 PDP 어드레스는 UMTS 패킷 도메인에서 사용되는 UE(111)의 IP 어드레스를 나타내며, 상기 PDP 컨텍스트 정보들을 구성하는 정보이다. 여기서, 상기 PDP 컨텍스트는 상기 GTP 터널의 각종 정보들을 저장하고 있으며, 상기 PDP 컨텍스트는 PDP 컨텍스트 ID로 관리된다. 그리고 상기 TI는 UE(111)와, UTRAN(113) 및 SGSN(115)에서 사용되며, GTP 터널들 각각을 구분하기 위해서 GTP 터널들 각각에 고유한 값으로 지정된다. 그리고 상기 TI와 상기 NSAPI는 유사한 개념으로 사용되나, 상기 TI는 UE(111)와, UTRAN(113) 및 SGSN(115)에서 사용되며, 상기 NSAPI는 UE(111)와, SGSN(115) 및 GGSN(119)에서 사용된다는 점에서 상이하다. 그리고, 상기 PDP 타입은 현재 상기 PDP 컨텍스트 활성화 요청 메시지를 통해 생성하고자 하는 GTP 터널의 종류, 즉 타입을 나타낸다. 여기서, 상기 GTP 터널의 종류는 인터넷 프로토콜(IP: Internet Protocol), 포인트 대 포인트 프로토콜(PPP: Point to Point Protocol)과, 모바일 IP(Mobile IP)등이 존재한다. 그리고 상기 접속 포인트 네트워크는 상기 GTP 터널을 생성 요청하는 UE(111)가 현재 접속하고자하는 서비스 네트워크의 접속 포인트를 나타낸다. 또한 상기 서비스 품질은 현재 생성되는 GTP 터널을 통해 전송되는 패킷 데이터의 품질을 나타낸다. 즉, 상기 서비스 품질이 높은 GTP 터널을 사용하는 패킷 데이터는 서비스 품질이 낮은 GTP 터널을 사용하는 패킷 데이터보다 우선 처리된다.

<49> 한편, 상기 PDP 컨텍스트 활성화 요청 메시지를 수신한 SGSN(115)는 UTRAN(113)로 무선 접속 베어러 셋업(Radio Access Bearer Setup) 메시지를 전송하여 상기 UTRAN(113)과 무선 접속 베어러를 설정하고(413단계), 또한 상기 UTRAN(113)은 상기 UE(111)로 무

선 접속 베어러 셋업 메시지를 전송하여 상기 UE(111)와 무선 접속 베어러를 설정한다(415단계). 이렇게, 상기 SGSN(115)과 UTRAN(113)간에, 또한 UTRAN(113)과 UE(111)간에 무선 접속 베어러가 설정됨에 따라 무선을 통한 패킷 데이터 전송에 필요한 자원(resource)이 할당이 완료된 것이다. 한편, 상기 도 4에 도시되어 있는 "Invoke Trace" 메시지를 설명하면 다음과 같다. 상기 UTRAN(113)에 추적(trace) 기능이 활성화되어 있을 경우 상기 SGSN(115)은 상기 Invoke Trace 메시지를 홈위치 등록기(도시하지 않음)나 운용 및 유지보수 센터(OMC: Operation and Maintenance Center, 도시하지 않음)로부터 얻은 추적(trace) 정보와 함께 상기 UTRAN(113)에 전달한다. 여기서, 상기 추적 기능은 데이터의 흐름을 추적하기 위한 용도로서 사용된다.

<50> 한편, 상기 UTRAN(113)과 무선 접속 베어러가 설정된 상태에서 상기 SGSN(115)은 GGSN(119)으로 PDP 컨텍스트 생성 요청(Create PDP Context Request) 메시지를 전송한다(417단계). 이때 상기 SGSN(115)과 GGSN(119) 사이에는 터널 종단 포인트 ID(TEID: Tunnel Endpoint ID)가 새롭게 설정되는데, 상기 터널 종단 포인트 ID는 GTP 터널을 사용하는 네트워크 노드들간에 패킷 데이터를 전송하기 위해 설정되는 것이다. 즉, 상기 SGSN(115)은 GGSN(119)의 터널 종단 포인트 ID를 기억하고 있으며, 상기 GGSN(119)은 상기 SGSN(115)의 터널 종단 포인트 ID를 기억하고 있다. 그래서, 상기 PDP 컨텍스트 생성 요청 메시지에는 상기 GGSN(199)이 상기 SGSN(115)로 패킷 데이터를 전송할 때 사용하여야 할 터널 종단 포인트 ID가 포함되어 있다.

<51> 상기 PDP 컨텍스트 생성 요청 메시지를 수신한 GGSN(199)은 상기 PDP 컨텍스트 생성 요청 메시지에 대한 PDP 컨텍스트 생성이 정상적으로 완료되면 상기 SGSN(115)로 PDP 생성 응답(Create PDP Context Response) 메시지를 전송한다(419단계). 이로써 상기

SGSN(115)과 GGSN(119)간에 GTP 터널 생성이 완료되는 것이며, 상기 GTP 터널 생성으로 인해 실제 패킷 데이터 전송이 가능해지는 것이다. 상기 PDP 생성 응답 메시지를 수신한 SGSN(115)은 상기 UE(111)로 PDP 활성화 허용(Activate PDP Context Accept) 메시지를 전송한다(421단계). 상기 UE(111)가 상기 PDP 활성화 허용 메시지를 수신함에 따라 상기 UE(111)와 UTRAN(113) 사이에 무선 채널(radio channel)이 생성되며, 결과적으로 상기 UTRAN(113)과, SGSN(115) 및 GGSN(119) 사이에 GTP 터널이 생성 완료된 것이다. 즉, 상기 UE(111)는 UE(111) 자신의 PDP 어드레스로 전달되는 모든 패킷 데이터들을 송수신하는 것이 가능하게 된다. 한편, 상기에서 설명한 PDP 컨텍스트 과정에서 생성된 GTP 터널은 하나의 PDP 컨텍스트와 일대일 대응하며, GTP 터널이 상이하면 PDP 컨텍스트가 상이함으로써 다른 터널 정보를 가지게 된다.

<52> 상기 도 4에서는 일반적인 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정, 즉 제1 PDP 컨텍스트 활성화 과정을 설명하였으며, 다음으로 도 5를 참조하여 제2 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 설명하기로 한다.

<53> 상기 도 5는 제2 PDP 컨텍스트 활성화에 따른 GTP 터널 생성 과정을 도시한 신호 흐름도이다.

<54> 먼저, 상기 제2 PDP 컨텍스트 활성화 과정은 이미 활성화되어 있는 제1 PDP 컨텍스트의 GTP 터널 정보를 그대로 재사용하여 GTP 터널을 새롭게 생성하는 과정을 의미하는 것이다. 즉, 상기 제2 PDP 컨텍스트 활성화 과정에 따라 생성되는 GTP 터널은 상기에서 설명한 바와 같이 제2 GTP 터널이라 하며, 상기 제2 GTP 터널은 상기 제1 PDP 컨텍스트 정보를 그대로 사용한다.

<55> 상기 도 5를 참조하면, UE(111)는 제2 GTP 터널 생성을 위해 SGSN(115)으로 제2 PDP 컨텍스트 활성화 요청(Activate Secondary PDP Context Request) 메시지를 전송한다(511단계). 상기 제2 PDP 컨텍스트 활성화 요청 메시지에 포함되는 파라미터(parameter)들로는 NSAPI와, Linked TI와, PDP 타입과, PDP 어드레스와, 접속 포인트 네트워크 및 서비스 품질 등이 있다. 여기서, 상기 제2 PDP 컨텍스트 활성화 요청 메시지는 상기 PDP 컨텍스트 활성화 요청 메시지와는 달리 Linked TI를 포함시켜 전송하는데 이는 이미 활성화되어 있는 제1 PDP 컨텍스트의 정보, 즉 제1 GTP 터널 정보를 그대로 사용하기 위한 것이다. 상기 도 4에서 설명한 바와 같이 TI는 UE(111)와, UTRAN(113) 및 SGSN(115) 간에서 GTP 터널을 구분하기 위해서 사용되는 것이므로, 상기 제1 GTP 터널과 동일한 정보를 사용하기 위해서 Linked TI를 사용하는 것이다.

<56> 한편, 상기 제2 PDP 컨텍스트 활성화 요청 메시지를 수신한 SGSN(115)는 UTRAN(113)로 무선 접속 베어러 셋업(Radio Access Bearer Setup) 메시지를 전송하여 상기 UTRAN(113)과 무선 접속 베어러를 설정하고(513단계), 또한 상기 UTRAN(113)은 상기 UE(111)로 무선 접속 베어러 셋업 메시지를 전송하여 상기 UE(111)와 무선 접속 베어러를 설정한다(515단계). 이렇게, 상기 SGSN(115)과 UTRAN(113)간에, 또한 UTRAN(113)과 UE(111)간에 무선 접속 베어러가 설정됨에 따라 무선을 통한 패킷 데이터 전송에 필요한 자원 할당이 완료된다.

<57> 한편, 상기 UTRAN(113)과 무선 접속 베어러가 설정된 상태에서 상기 SGSN(115)은 GGSN(119)으로 PDP 컨텍스트 생성 요청(Create PDP Context Request) 메시지를 전송한다(517단계). 이때 상기 SGSN(115)는 상기 생성하고자 하는 GTP 터널이 제2 GTP 터널임을 나타내기 위해서 제1 NSAPI(Primary NSAPI)를 전송하는데, 상기 제1 NSAPI 값

은 이미 활성화되어 있는 제1 PDP 컨텍스트의 정보와 일대일로 대응된다. 이는 상기 제1 NSAPI 값을 참조하여 제1 PDP 컨텍스트 정보를 사용할 수 있기 때문이다. 또한 상기 SGSN(115)은 상기 PDP 컨텍스트 생성 요청 메시지에 TFT를 포함하여 전송한다. 그 이유는 상기 제1 GTP 터널과 제2 GTP 터널들을 구분하기 위함이다. 즉, 상기 제1 GTP 터널에는 TFT가 저장되어 있지 않으며, 상기 제2 GTP 터널들에만 TFT가 저장되어 있기 때문이다. 그리고 상기 제1 GTP 터널 생성과 마찬가지로 상기 SGSN(115)과 GGSN(119) 사이에는 터널 종단 포인트 ID가 새롭게 설정되는데, 상기 터널 종단 포인트 ID는 GTP 터널을 사용하는 네트워크 노드들간에 패킷 데이터를 전송하기 위해 설정되는 것이다. 즉, 상기 SGSN(115)은 GGSN(119)의 터널 종단 포인트 ID를 기억하고 있으며, 상기 GGSN(119)은 상기 SGSN(115)의 터널 종단 포인트 ID를 기억하고 있다. 그래서, 상기 PDP 컨텍스트 생성 요청 메시지에는 상기 GGSN(199)이 상기 SGSN(115)로 패킷 데이터를 전송할 때 사용하여야 할 터널 종단 포인트 ID가 포함되어 있다.

<58> 상기 PDP 컨텍스트 생성 요청 메시지를 수신한 GGSN(199)은 상기 PDP 컨텍스트 생성 요청 메시지에 대한 PDP 컨텍스트 생성이 정상적으로 완료되면 상기 SGSN(115)로 PDP 생성 응답(Create PDP Context Response) 메시지를 전송한다(519단계). 이로써 상기 SGSN(115)과 GGSN(119)간에 제2 GTP 터널 생성이 완료되는 것이며, 상기 제2 GTP 터널 생성으로 인해 실제 패킷 데이터 전송이 가능해지는 것이다. 상기 PDP 생성 응답 메시지를 수신한 SGSN(115)은 상기 UE(111)로 PDP 활성화 허용(Activate PDP Context Accept) 메시지를 전송한다(521단계). 상기 UE(111)가 상기 PDP 활성화 허용 메시지를 수신함에 따라 상기 UE(111)와 UTRAN(113) 사이에 무선 채널이 생성되며, 상기 UTRAN(113)과, SGSN(115) 및 GGSN(119) 사이에 제2 GTP 터널이 생성 완료된 것이다. 즉, 상기 UE(111)

는 UE(111) 자신의 PDP 어드레스로 전달되는 모든 패킷 데이터들을 송수신하는 것이 가능하게 된다. 한편, 상기에서 설명한 PDP 컨텍스트 과정에서 생성된 제2 GTP 터널 역시 하나의 PDP 컨텍스트와 일대일 대응한다.

<59> 다음으로 상기 도 3에서 설명한 TFT 연산 코드에 따른 TFT 처리를 설명하기로 하며, 먼저 도 6을 참조하여 새로운 TFT를 생성하는 과정을 설명하기로 한다.

<60> 상기 도 6은 새로운 TFT 생성을 위한 TFT 정보들을 개략적으로 도시한 도면이다.

<61> 먼저, 상기 도 3에서 설명한 바와 같이 TFT의 TFT 연산 코드가 "001"로 설정되어 있을 경우, 새로운 TFT를 생성하게 된다. 한편, 상기 도 6에 도시되어 있는 "0" 영역은 스페어 비트(spare bit)로서 그 용도가 아직 정해지지 않은 미정 영역으로서, 일반적으로 "0"으로 설정한다. 그리고 상기 도 6에서는 패킷 필터 리스트 영역을 좀 더 세분화하여 상세하게 설명하기로 한다. 상기 도 6에서, 먼저 패킷 필터 ID(packet filter identifier)는 상기 TFT내에 설정되어 있는 다수개의 패킷 필터들중 해당 패킷 필터를 구분하기 위해서 사용된다. 상기에서 설명한 바와 같이 TFT에 설정될 수 있는 최대 패킷 필터수는 일 예로 최대 8개로 가정하였기 때문에, 상기 패킷 필터 ID 역시 최대 8개로 표현될 수 있다. 상기 도 6에서는 0~2 비트로 표현하며, 나머지 4~7비트는 스페어 비트로 설정하였다.

<62> 다음으로 패킷 필터 평가 순위(evaluation precedence)는 상기 TFT에 설정되어 있는 모든 패킷 필터들간에 적용되는 순서를 나타낸다. 즉, 외부 네트워크로부터 입력되는 패킷 데이터에 대해서 어느 패킷 필터부터 적용할지의 순서를 나타내는 것으로서, 상기 패킷 필터 평가 순위 값이 작으면 작을수록 상기 외부 네트워크로부터 입력되는 패킷 데이터에 대해 적용되는 순서가 빠르게 된다. 상기 외부 네트워크로부터 패킷 데이터가 수

신되면 상기 GGSN(119)에 저장되어 있는 TFT 패킷 필터들중 상기 패킷 필터 평가 순위 값이 가장 작은 패킷 필터부터 상기 수신되는 패킷 데이터에 적용하며, 상기 가장 작은 패킷 필터 평가 순위 값을 가지는 패킷 필터가 상기 수신된 패킷 데이터의 헤더(header)가 매칭(matching)되지 않을 경우 상기 패킷 필터 평가 순위 값이 그 다음으로 작은 패킷 필터에 상기 수신된 패킷 데이터를 적용시킨다. 그리고 상기 패킷 필터 컨텐츠 길이(Length of Packet filter contents)는 해당 패킷 필터의 컨텐츠 길이를 나타낸다.

<63> 마지막으로 패킷 필터 컨텐츠는 패킷 필터 컴퍼넌트 타입 ID(packet filter component type identifier)를 포함하며, 그 길이가 가변적이다. 상기 패킷 필터 컨텐츠의 길이가 가변적인 이유는 상기 패킷 필터의 길이가 각각 다르며, 또한 TFT에 설정되는 패킷 필터들의 개수가 상황에 따라 가변적이기 때문이다. 그리고, 상기 패킷 필터 컴퍼넌트 타입 ID는 한 번 사용된 후에는 어떤 패킷 필터에도 사용되는 것이 불가능하며, 같은 TFT 내에서 IPv4(IP version 4) source address type과 IPv6(IP version 6) source address type을 혼용해서 패킷 필터를 구성할 수 없다. 그리고 단일 데스티네이션 포트 타입(single destination port type)과 데스티네이션 포트 범위 타입(destination port range type)도 상기 패킷 필터에서 혼용하여 구성할 수 없다. 또한 단일 소스 포트 타입(single source port type)과 소스 포트 타입 범위 타입(source port range type)도 상기 패킷 필터에서 혼용하여 구성할 수 없다. 상기에서 설명한 바와 같은 패킷 필터 컴퍼넌트 타입들과 해당 패킷 필터 컴퍼넌트 타입 ID를 하기 표 2에 나타내었다.

<64>

【표 2】

Bits (76543210)	내용
0001 0000	IPv4 source address type
0010 0000	IPv6 source address type
0011 0000	Protocol identifier/Next header type
0100 0000	Single destination port type
0100 0001	Destination port range type
0101 0000	Single source port type
0101 0001	Source port range type
0110 0000	Security parameter index type
0111 0000	Type of service / Traffic class type
1000 0000	Flow label type
All other values	예약됨

<65> 상기 표 2에 나타낸 바와 같이, 하나의 패킷 필터에 다수개의 패킷 필터 컴퍼넌트들이 구성될 수 있다. 그런데, 현재 UMTS 통신 시스템에서는 제안하고 있는 모든 패킷 필터 타입을 사용하지는 않는다. 일 예로, TCP/UDP 포트 범위(TCP/UDP port range)는 패킷 필터 컴퍼넌트로 사용하지만, TCP/UDP 포트(TCP/UDP port) 각각을 패킷 필터 컴퍼넌트로 사용하지는 않는다. 그리고, 패킷 필터 컴퍼넌트는 패킷 필터에 다수개로 구성될 수 있다. 예를 들면 종단 장치(TE: Terminal Equipment)가 ::172.168.8.0/96에서 TCP 포트 범위를 4500~5000으로하는 IPv6 패킷 데이터를 분류할 수 있으며,

<66> packet filter identifier = 1;

<67> IPv6 Source Address = {::172.168.8.0[FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:0:0]};

<68> TCP에 대한 Protocol Number = 6;

<69> Destination Port range= 4500,5000;

<70> 와 같이 패킷 필터를 구성할 수 있다. 이런 식으로 다수개의 파라미터들을 사용하여 패킷 데이터를 분류하는 것이 멀티-필드 분류(Multi-field classification)라고 하며, 하기에서 패킷 필터 컴퍼넌트 타입들을 설명한다.

<71> 첫 번째로, IPv4 소스 어드레스 타입(IPv4 source address type)을 설명하기로 한다.

<72> 상기 IPv4 소스 어드레스 타입으로 설정된 패킷 필터 컨텐츠는 4옥텟(oct) 크기를 가지는 IPv4 어드레스 필드와, 4옥텟의 IPv4 어드레스 마스크(mask) 필드로 구성되며, 상기 IPv4 어드레스 필드가 상기 IPv4 어드레스 마스크 필드보다 먼저 전달된다. 여기서, 상기 IPv4 어드레스는 32비트로 표현되며, 일 예로 10.2.10.3과 같이 표현된다.

<73> 상기 IPv4 어드레스 필드는 접속 포인트 명(APN: Access Point Name, 이하 "APN"이라 칭하기로 한다)등의 서비스 네트워크에 접속하기 위해 사용되는 제2 PDP 컨텍스트 요청 메시지로 전달되는 TFT에는 설정하지 못하는 경우가 존재한다. 즉, UE(111)는 초기에 제2 PDP 컨텍스트를 활성화하면서 최초로 접속하는 서비스 네트워크에 대해서는 도메인 네임 서비스(DNS: Domain Name Service, 이하 "DNS"라 칭하기로 한다) 서버(server)를 통해서 실제 IP 어드레스를 수신하게 된다. 이와 같은 경우에는 이미 제2 PDP 컨텍스트 활성화 메시지를 전달하기 위해서 대기중이기 때문에 설정되는 TFT의 패킷 필터 컨텐츠를 변경하는 것이 불가능하다. 물론 상기 최초 접속 이후의 다음번 접속부터는 상기 UE(111)가 상기 DNS 서버로부터 수신한 해당 서비스의 IP 어드레스를 알고 있기 때문에 상기 설정되어 있는 TFT 패킷 필터의 컨텐츠로서 상기 IPv4 소스 어드레스 타입을 사용할 수 있다. 한편, 상기 UE(111)가 새로운 서비스 네트워크로 최초 접속하는 것이 아니라 다른 이동국과 통신을 하기 위해서 상기 제2 PDP 컨텍스트 활성화 요청 메시지를 전

송할 경우에는 상기 TFT에 IPv4 소스 어드레스 타입을 패킷 필터 컨텐츠로 사용할 수 있음은 물론이다.

<74> 두 번째로, IPv6 소스 어드레스 타입(IPv6 source address type)에 대해서 설명하기로 한다. 상기 IPv6 소스 어드레스 타입은 16옥텟(oct)의 IPv6 어드레스 필드와 16옥텟의 IPv6 어드레스 마스크 필드로 구성되며, 상기 IPv6 어드레스 필드가 상기 IPv6 어드레스 마스크 필드보다 먼저 전달된다. 여기서, 상기 IPv6 어드레스는 128비트로 표현되며, 상기 IPv6 어드레스를 사용할 경우 IPv4 어드레스에 비해서 2^{96}배만큼의 가입자들을 더 수용할 수 있다. 이렇게 상기 IPv4 어드레스에 비해서 굉장히 많은 가입자들을 추가적으로 수용할 수 있기 때문에 IPv6 어드레스의 사용이 증가되고 있다.

<75> 여기서 도 7을 참조하여 상기 IPv6 어드레스의 구조를 살펴보기로 한다.

<76> 상기 도 7은 일반적인 IPv6 어드레스 구조를 개략적으로 도시한 도면이다.

<77> 상기 도 7을 참조하면, 상기 IPv6 어드레스는 128비트로 표현되며, 실제 노드(node) 주소가 상기 128비트로 표현되는 것이다.

<78> 그러나, 상기 IPv6 어드레스의 가장 큰 단점은 어드레스 길이가 너무 길다는 것이다. 일 예로, 상기 IPv4 어드레스는 10.2.10.3으로 표현되지만, 상기 IPv6 어드레스는 ABCD:1234:EF12:5678:2456:9ABC로 표현된다. 이와 같이 IPv6 어드레스는 그 길이가 너무 길어 가입자들이 외우기에도 난이하고, 또한 그 연산 처리에 있어서도 128비트를 사용해야하기 때문에 로드 발생 및 소모되는 비용의 추가 등의 문제점이 있다.

<79> 세 번째로, 프로토콜 ID(Protocol identifier)/ Next header 타입에 대해서 설명하기로 한다. 상기 프로토콜 ID/Next header 타입은 1옥텟(oct)의 프로토콜 ID, 일 예로

IPv4나 Next header 타입, 일 예로 IPv6으로 구성된다. 네 번째로 단일 데스티네이션 포트 타입(Single destination port type)은 2옥텟(oct)의 데스티네이션 포트 넘버(destination port number)로 구성되며, 상기 단일 데스티네이션 포트 타입은 IP 헤더(header)의 프로토콜 필드값에 따라 UDP 포트값 혹은 TCP 포트값이 될 수 있다. 다섯 번째로 데스티네이션 포트 범위 타입(Destination port range type)은 2옥텟(oct)의 데스티네이션 포트 넘버(destination port number)의 최소값과 2옥텟(oct)의 데스티네이션 포트 넘버 최대값으로 구성되며, 상기 데스티네이션 포트 범위 타입은 IP 헤더의 프로토콜 필드 값에 따라 UDP 포트 혹은 TCP 포트의 범위가 될 수 있다.

<80> 여섯 번째로, 단일 소스 포트 타입(Single source port type)은 2옥텟(oct)의 소스 포트 넘버(source port number)로 구성되며, IP 헤더의 프로토콜 필드 값에 따라 UDP 포트 혹은 TCP 포트 값이 될 수 있다. 일곱 번째로, 소스 포트 범위 타입(Source port range type)은 2옥텟(oct)의 소스 포트 넘버(source port number)의 최소값과 2옥텟(oct)의 소스 포트 넘버 최대값으로 구성되며, IP 헤더의 프로토콜 필드 값에 따라 UDP 포트 혹은 TCP 포트의 범위가 될 수 있다. 여덟 번째로, 보안성 파라미터 인덱스 타입(Security parameter index type)은 4옥텟(oct)의 IPSec security parameter Index(SPI)로 구성된다. 아홉 번째로, 서비스 타입(Type of service)/트래픽 클래스 타입(Traffic class type)은 1옥텟(oct)의 IPv4 서비스 타입(Type of service(IPv4))/IPv6 트래픽 클래스(Traffic class (IPv6))과, 1옥텟(oct)의 IPv4 서비스 마스크 타입(Type of service mask (IPv4))/ IPv6 트래픽 클래스 마스크(Traffic class mask (IPv6))로 구성된다. 마지막으로 플로우 라벨 타입(Flow lavel type)은 3옥텟(oct)의 IPv6 플로우 라벨로 구성

되며, 첫 번째 옥텟의 4~7비트는 스페어 필드(spare field)이며, 나머지 20비트에 IPv6 플로우 라벨이 포함되어 있다.

<81> 상기 도 6에서는 TFT 연산 코드가 "001"인 경우, 즉 새로운 TFT를 생성하는 과정을 설명하였으며, 다음으로 도 8을 참조하여 TFT 연산 코드가 "010"인 경우, 즉 저장중인 TFT를 삭제하는 과정과, TFT 연산 코드가 "011"인 경우, 즉 저장중인 TFT에 패킷 필터를 첨가하는 과정과, TFT 연산 코드가 "100"인 경우, 즉 저장중인 TFT에 패킷 필터를 대체하는 과정을 설명하기로 한다.

<82> 상기 도 8은 저장되어 있는 TFT를 삭제하거나 저장되어 있는 TFT에 패킷 필터를 첨가하거나 혹은 패킷 필터 대치를 하기 위한 TFT 정보들을 개략적으로 도시한 도면이다.

<83> 상기 도 8을 참조하면, 첫 번째로, TFT를 삭제할 경우에는 패킷 필터 리스트 영역은 별도로 상관할 필요없이 TFT 연산 코드를 확인한 후 상기 TFT 연산 코드 값이 미리 설정한 TFT 삭제를 나타내는 값, 즉 "010"일 경우 GGSN(119)에 저장되어 있는 TFT들 중 상기 삭제하고자 하는 TFT 타입과 동일한 TFT를 상기 GGSN(119)에서 삭제한다. 두 번째로, 저장되어 있는 TFT에 패킷 필터를 추가할 경우에는 상기에서 설명한 TFT를 삭제하는 경우와 동일한 정보들이 사용되며, 해당 패킷 필터 리스트의 컨텐츠를 상기 저장되어 있는 TFT에 첨가한다. 세 번째로, 저장되어 있는 TFT의 패킷 필터를 대치할 경우에 사용하는 정보 역시 상기 TFT를 삭제하는 경우 및 TFT에 패킷 필터를 추가하는 경우와 동일한 정보가 사용되며, 해당 패킷 필터 리스트의 내용을 상기 저장되어 있는 TFT의 패킷 필터를 삭제한 후 대치한다.

<84> 상기 도 8에서는 TFT 연산 코드가 "010"인 경우, 즉 저장중인 TFT를 삭제하는 과정과, TFT 연산 코드가 "011"인 경우, 즉 저장중인 TFT에 패킷 필터를 첨가하는 과정과, TFT 연산 코드가 "100"인 경우, 즉 저장중인 TFT에 패킷 필터를 대체하는 과정을 설명하였으며, 다음으로 도 9를 참조하여 TFT 연산 코드가 "101"인 경우, 즉 저장중인 TFT 패킷 필터를 삭제하는 과정을 설명하기로 한다.

<85> 상기 도 9는 저장되어 있는 TFT 패킷 필터를 삭제하기 위한 TFT 정보들을 개략적으로 도시한 도면이다.

<86> 상기 도 9에 도시되어 있는 바와 같이 저장되어 있는 TFT에서 패킷 필터를 삭제할 경우에는 패킷 필터 리스트는 상관없이 패킷 필터 ID만 고려하게 된다. 상기 GGSN(119)는 저장되어 있는 TFT의 패킷 필터들에서 UE(111)로부터 전달받은 상기 TFT 정보의 패킷 필터 ID에 해당하는 패킷 필터를 삭제한다. 상기 도 9에서는 제1패킷 필터부터 제N 패킷 필터까지 N개의 패킷 필터들을 TFT에서 삭제하는 경우이다.

<87> 다음으로 도 10을 참조하여 TFT 패킷 필터링 과정을 설명하기로 한다.

<88> 상기 도 10은 일반적인 UMTS 코어 네트워크에서 TFT 패킷 필터링 과정을 개략적으로 도시한 도면이다.

<89> 먼저, 상기 도 10을 설명함에 있어서 TFT 패킷 필터링을 설명할 때 설명상 편의를 위하여 각 TFT가 단 한 개의 패킷 필터들만을 가진 경우를 가정하여 설명하기로 한다. UMTS 코어 네트워크(200)의 GGSN(119)에는 총 4개의 TFT들이 저장되어 있으며, 상기 4개의 TFT 필터들 각각은 한 개의 패킷 필터를 가진다. 또한, 상기 TFT가 4개 저장되어 있다는 것은 상기 GGSN(119)은 SGSN(115)과 5개의 GTP 터널, 즉 제1 PDP 컨텍스트를 위한

1개의 제1 GTP 터널과, 제2 PDP 컨텍스트들을 위한 4개의 제2 GTP 터널들을 구비하며, 상기 5개의 GTP 터널들이 같은 PDP 컨텍스트를 공유하게 된다. 그리고 상기 총 5개의 GTP 터널들은 TFT에 의해서만 구분이 된다.

<90> 외부 네트워크, 일 예로 인터넷(121)으로부터 입력되는 패킷 데이터가 상기 4개의 TFT들을 통해 패킷 필터링이 성공하지 못할 경우에는 상기 인터넷(121)으로부터 입력된 패킷 데이터는 오직 제1 PDP 컨텍스트(제1 GTP 터널)만을 통해 SGSGN(115)로 전송된다. 일 예로 상기 인터넷(121)에서 입력된 패킷 데이터가 서비스 타입(TOS: Type Of Service)이 0x30, 프로토콜이 TCP, 소스 어드레스가 1.1.1.1, 데스티네이션 어드레스가 2.2.2.2, 소스 포트가 200, 데스티네이션 포트가 50인 경우를 가정하면, 상기 입력된 패킷 데이터는 TFT 1 및 TFT2 까지는 패킷 필터 컨텐츠에 부합되지 않아 패킷 필터링이 수행되지 않으며 TFT 3의 패킷 필터 컨텐츠에 부합하여 패킷 필터링되고, 상기 부합하는 TFT3에 해당하는 GTP 터널을 통해서 상기 SGSN(115)로 전달된다. 여기서, 상기 인터넷(121)에서 입력된 패킷 데이터가 TFT 1 및 TFT 2에서 필터링되지 못하는 이유는 상기 TFT 1 패킷 필터 컨텐츠인 소스 어드레스는 3.3.3.3이므로 상기 입력된 패킷 데이터의 소스 어드레스 1.1.1.1과 일치하지 않으며, 상기 TFT 2 패킷 필터 컨텐츠인 프로토콜은 ICMP이므로 상기 입력된 패킷 데이터의 프로토콜 TCP와 일치하지 않기 때문이다. 그리고 상기 TFT 3에서 필터링되는 이유는 상기 TFT 3 패킷 필터 컨텐츠인 서비스 타입이 0x30이므로 상기 입력된 패킷 데이터의 서비스 타입 0x30과 일치하기 때문이다.

<91> 상기에서 설명한 바와 같이 TFT는 제2 PDP 컨텍스트 활성화 과정에서 PDP 컨텍스트(GTP 터널)와 항상 연관되어 생성된다. 상기 TFT는 PDP 컨텍스트 활성화 과정에서 생성된 PDP 컨텍스트를 UE(111)가 PDP 컨텍스트 수정 과정(UE-Initiated PDP Context

Modification Procedure)을 통해서 추가/수정/삭제가 가능하며, 상기에서 설명한 바와 같이 하나의 PDP 컨텍스트는 오직 하나의 TFT만을 가질 수 있다. 여기서, 상기 UE(111)가 새로운 TFT를 생성하거나 혹은 상기 GGSN(119)에 저장되어 있는 TFT를 수정하고자 할 경우, 상기 TFT는 적어도 하나 이상의 유효한 패킷 필터를 저장하고 있어야만 한다. 만약 상기 저장되어 있는 TFT에 유효한 패킷 필터가 존재하지 않을 경우 상기 UE(111)의 PDP 컨텍스트 수정 과정(MS-Initiated PDP Context Modification Procedure)은 실패하게 되며, 상기 GGSN은 상기 UE(111)로 상기 TFT를 위한 상기 UE(111) 자신의 PDP 컨텍스트 수정 과정(MS-Initiated PDP Context Modification Procedure)이 실패함을 나타내는 오류 코드를 전송한다. 또한, 상기 TFT는 TFT와 관련된 PDP 컨텍스트가 비활성화되면 삭제된다.

<92> 한편, 상기에서 설명한 IP 어드레스에 대해서 구체적으로 설명하면 다음과 같다.

<93> 상기 IP 어드레스는 그 버전에 따라서 IPv4 어드레스와 IPv6 어드레스로 구분되는데, 상기 IPv4 어드레스를 사용하는 네트워크를 "IPv4 네트워크"라 칭하기로 하며, IPv6 어드레스를 사용하는 네트워크를 "IPv6 네트워크"라 칭하기로 한다. 상기 UMTS 통신 시스템은 IPv4 네트워크와 IPv6 네트워크간에 IP 통신이 가능하도록 IPv4 삽입 IPv6 어드레스(이하 "IPv4 embedded IPv6 어드레스"라 칭하기로 한다)를 사용한다. 여기서, 상기 IPv4 embedded IPv6 어드레스는 IPv4 호환(compatible) IPv6(이하 "IPv4 compatible IPv6"라 칭하기로 한다) 어드레스와, IPv4 매핑(mapped) IPv6(이하 "IPv4 mapped IPv6"라 칭하기로 한다) 어드레스를 정의하고 있다. 상기 IPv4 compatible IPv6 어드레스와 IPv4 mapped IPv6 어드레스를 설명하면 다음과 같다.

<94> (1) IPv4 compatible IPv6 어드레스

<95> 상기 IPv4 compatible IPv6 어드레스는 상대방 네트워크가 IPv6 어드레스를 지원하며, 상대방, 즉 데스티네이션의 IPv4 어드레스를 알고 있고, IPv6 네트워크를 통해 통신하고자 할 경우 선택적으로 사용되는 어드레스이다. 그러면 여기서 상기 IPv4 compatible IPv6 어드레스 구조를 도 11을 참조하여 설명하기로 한다.

<96> 상기 도 11은 일반적인 IPv4 compatible IPv6 어드레스 구조를 개략적으로 도시한 도면이다.

<97> 상기 도 11을 참조하면, 기본적으로 상기 IPv4 compatible IPv6 어드레스는 IPv6 어드레스이기 때문에 128비트로 표현되며, 하위 32비트가 IPv4 어드레스가 삽입된다. 즉, 상기 하위 32비트에는 데스티네이션의 IPv4 어드레스 32 비트가 그대로 삽입되며, 나머지 상위 96비트에는 모두 0이 삽입된다.

<98> 그러면 여기서 도 12를 참조하여 상기 IPv4 compatible IPv6 어드레스가 사용되는 네트워크 구조를 설명하기로 한다.

<99> 상기 도 12는 IPv4 compatible IPv6 어드레스가 사용되는 네트워크 구조를 개략적으로 도시한 도면이다.

<100> 상기 도 12를 참조하면, 먼저 네트워크(1211)와 네트워크(1213)는 IPv4 어드레스 및 IPv6 어드레스 모두를 사용하는 네트워크이며, 상기 네트워크(1211)에서 전송하고자 하는 패킷 데이터의 데스티네이션의 어드레스가 IPv4 어드레스일 경우 상기 네트워크(1211)는 상기 IPv4 어드레스를 구성하는 32비트를 상기 도 11에서 설명한 바와 같이 IPv4 compatible IPv6 어드레스의 하위 32비트에 삽입하여 상기 네트워크(1213)로 전송한다. 그러면 상기 네트워크(1213)는 상기 네트워크(1211)에서 전송한 IPv4 compatible

IPv6 어드레스의 패킷 데이터를 수신하고, 상기 네트워크(1213)는 상기 IPv4 compatible IPv6 어드레스의 하위 32비트의 IPv4 어드레스를 검출한다. 여기서, 상기 IPv4 어드레스는 글로벌하게 유일해야하는데, 이는 IPv4 어드레스만으로도 유일성이 보장되어야 한다는 것이다. 여기서, 상기 IPv4 compatible IPv6 어드레스는 다음과 같이 표현된다.

<101> 0:0:0:0:0:0:165.213.138.35 → ::165.213.138.35

<102> 이렇게, IPv4 compatible IPv6 어드레스는 하위 32비트에 IPv4 어드레스가 삽입되어 있는 형태를 가지며, 상기에서 설명한 바와 같이 상기 IPv4 compatible IPv6 어드레스 역시 글로벌하게 유일한 어드레스가 된다.

<103> (2) IPv4 mapped IPv6 어드레스

<104> 상기 IPv4 mapped IPv6 어드레스는 상대방 네트워크가 IPv6 어드레스를 지원하지 않지만, IPv6 어드레스를 가지고 통신을 수행해야할 경우 선택적으로 사용되는 어드레스이다. 그러면 여기서 상기 IPv4 mapped IPv6 어드레스 구조를 도 13을 참조하여 설명하기로 한다.

<105> 상기 도 13은 일반적인 IPv4 mapped IPv6 어드레스 구조를 개략적으로 도시한 도면이다.

<106> 상기 도 13을 참조하면, 기본적으로 상기 IPv4 mapped IPv6 어드레스는 IPv6 어드레스이기 때문에 128비트로 표현되며, 하위 32비트가 IPv4 어드레스가 삽입된다. 즉, 상기 하위 32비트에는 데스티네이션의 IPv4 어드레스 32 비트가 그대로 삽입되며, 상기 32비트의 IPv4 어드레스가 삽입된 바로 인접 상위 16비트에 1이 삽입되며, 나머지 상위 80비트에는 모두 0이 삽입된다.

<107> 그러면 여기서 도 14를 참조하여 상기 IPv4 mapped IPv6 어드레스가 사용되는 네트워크 구조를 설명하기로 한다.

<108> 상기 도 14는 IPv4 mapped IPv6 어드레스가 사용되는 네트워크 구조를 개략적으로 도시한 도면이다.

<109> 상기 도 14를 참조하면, 먼저 네트워크(1411)는 IPv4 어드레스 및 IPv6 어드레스 모두를 사용하는 네트워크이며, 네트워크(1413)는 IPv4 어드레스만을 사용하는 네트워크이다. 상기 네트워크(1411)에서 전송하고자하는 패킷 데이터의 데스티네이션의 어드레스가 IPv4 어드레스일 경우 상기 네트워크(1411)는 상기 IPv4 어드레스를 구성하는 32비트를 상기 도 13에서 설명한 바와 같이 IPv4 mapped IPv6 어드레스 어드레스의 하위 32비트에 삽입하여 상기 네트워크(1413)로 전송한다. 그러면 상기 네트워크(1413)는 상기 네트워크(1411)에서 전송한 IPv4 mapped IPv6 어드레스의 패킷 데이터를 수신하고, 상기 네트워크(1413)는 상기 IPv4 mapped IPv6 어드레스의 하위 32비트의 IPv4 어드레스를 검출한다. 여기서, 상기 IPv4 mapped IPv6 어드레스는 다음과 같이 표현된다.

<110> 0:0:0:0:FFFF:165.213.138.35 → ::FFFF:165.213.138.35

<111> 이렇게, IPv4 mapped IPv6 어드레스는 하위 32비트에 IPv4 어드레스가 삽입되어 있는 형태를 가지며, 상기에서 설명한 바와 같이 상기 IPv4 compatible IPv6 어드레스와 달리 상기 IPv4 어드레스가 삽입되는 32비트 인접 상위 16비트에 0xFFFF가 삽입되어 있다.

<112> 상기에서 설명한 TFT 패킷 필터의 컴퍼넌트 타입들중 IPv4 source address는 IPv4 어드레스를 사용하는 32비트 어드레스를 나타낸다. 현재 이동 통신 시스템의 가입자수는

기하급수적으로 증가하고 있으며, 이렇게 기하급수적으로 증가하는 가입자들에 정상적인 IP 어드레스를 할당하기 위해서는 IPv6 어드레스의 사용이 상용화될 것이다. 그래서, 상기 IPv6 어드레스를 가지는 패킷 데이터를 필터링하기 위한 TFT 패킷 필터의 컴퍼넌트 타입이 제안되었다. 그러나, 상기 IPv6 어드레스는 상기에서 설명한 바와 같이 128비트로 표현되기 때문에 IPv4 어드레스를 표현하기 위한 32비트에 비해서 비트 연산면에서 엄청난 로드를 발생하게 된다.

<113> 즉, 상기에서 설명한 바와 같이 외부 네트워크에서 GGSN(119)으로 입력되는 패킷 데이터들은 상기 GGSN(119)에 저장되어 있는 TFT를 통해 패킷 필터링되고, 상기 TFT를 통한 패킷 필터링은 상기 TFT내에 저장되어 있는 적어도 하나 이상의 패킷 필터들에 대해서 패킷 필터 평가 순위가 가장 작은 값을 가지는 패킷 필터부터 순차적으로 수행된다. 일 예로 상기 GGSN(119)에 5개의 TFT들이 저장되어 있고, 상기 5개의 TFT들이 각각 4개의 패킷 필터들을 저장하고 있을 경우 외부 네트워크, 즉 인터넷(121)으로부터 입력되는 패킷 데이터는 상기 5개의 TFT에 대해서 첫 번째 TFT부터 4개의 패킷 필터들에 대해 패킷 필터링을 수행하고, 패킷 필터링이 성공되지 못하였을 경우 두 번째 TFT순서로 패킷 필터링을 수행하여 상기 외부 네트워크로부터 입력된 패킷 데이터에 대해서 패킷 필터링을 수행하게 된다. 이렇게 유입된 패킷 데이터에 대한 패킷 필터링이 성공될 때까지 상기 IPv6 어드레스의 128비트 연산은 상기 GGSN(119)에 저장되는 TFT들의 개수가 급증할 경우 및 외부 네트워크(121)로부터 입력되는 패킷 데이터 양이 급증할 경우 상기 패킷 필터링의 성능 손실을 가져오며, 이런 패킷 필터링 성능 손상은 UMTS 코어 네트워크에 치명적으로 작용할 수 있다는 문제점이 발생한다.

【발명이 이루고자 하는 기술적 과제】

<114> 따라서, 본 발명의 이동 통신 시스템에서 IP 어드레스 버전에 따라 트래픽 플로우 탬플릿 패킷 필터링을 수행하는 장치 및 방법을 제공함에 있다.

<115> 본 발명의 다른 목적은 이동 통신 시스템에서 서로 다른 버전을 가지는 IP 어드레스에서 공통적으로 사용되는 영역을 사용하여 트래픽 플로우 탬플릿 패킷 필터링을 수행하는 장치 및 방법을 제공함에 있다.

<116> 본 발명의 또 다른 목적은 이동 통신 시스템에서 입력되는 패킷 데이터들의 IP 어드레스 버전에 따라 최소 패킷 필터링 계산량을 제공하는 트래픽 플로우 탬플릿 패킷 필터링을 수행하는 장치 및 방법을 제공함에 있다.

<117> 상기한 목적들을 달성하기 위한 본 발명의 제1실시예에 따른 장치는; 제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 장치에 있어서, TFT를 수신하고, 상기 수신된 TFT 정보가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하여 새로운 TFT 정보를 생성하도록 제어하는 제어기와, 상기 수신된 TFT 정보를 상기 새로운 TFT 정보로 저장하는 메모리를 포함함을 특징으로 한다.

<118> 상기한 목적들을 달성하기 위한 본 발명의 제2실시예에 따른 장치는; 제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을

포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 장치에 있어서, 소스 IP 어드레스가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하여 TFT 정보를 생성하고, 상기 생성한 TFT 정보를 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 사용자 단말기와, 상기 사용자 단말기로부터 수신한 TFT 정보를 저장하고, 이후 수신되는 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하고, 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 GGSN을 포함함을 특징으로 한다.

<119> 상기한 목적들을 달성하기 위한 본 발명의 제1실시예에 따른 방법은; 제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서, TFT를 수신하면, 상기 TFT 정보가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1 버전 IP 어드레스의 제1비트들을 추출하는 과정과, 상기 추출한 제1버전 IP 어드레스의 제1비트들을 새로운 TFT 정보로 생성하는 과정과, 이후 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스

를 나타내는 제1비트들을 추출하는 과정과, 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 과정을 포함함을 특징으로 한다.

<120> 상기한 목적들을 달성하기 위한 본 발명의 제2실시예에 따른 방법은; 제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서, 사용자 단말기는 소스 IP 어드레스가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하는 과정과, 상기 사용자 단말기는 추출한 제1 버전 IP 어드레스의 제1비트들을 패킷 필터 컨텐츠로 생성하고, 상기 패킷 필터 컨텐츠를 포함하는 TFT 정보를 생성하여 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 과정과, 상기 GGSN은 상기 TFT 정보를 저장하고, 이후 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하는 과정과, 상기 GGSN은 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 과정을 포함함을 특징으로 한다.

【발명의 구성 및 작용】

<121> 이하, 본 발명에 따른 바람직한 실시예를 첨부한 도면을 참조하여 상세히 설명한다. 하기의 설명에서는 본 발명에 따른 동작을 이해하는데 필요한 부분만이 설명되며 그

이외 부분의 설명은 본 발명의 요지를 흩트리지 않도록 생략될 것이라는 것을 유의하여야 한다.

<122> 도 15는 본 발명의 실시예에서의 기능을 수행하기 위한 UMTS 네트워크 구조를 개략적으로 도시한 도면이다.

<123> 상기 도 15를 참조하면, 먼저 UMTS 네트워크는 인터넷 프로토콜(IP: Internet Protocol, 이하 "IP"라 칭하기로 한다) 버전(version) 6(이하, "IPv6"라 칭하기로 한다) 어드레스(address)를 사용하는 IPv6 네트워크(1500)와, IP 버전 4(이하, "IPv4"라 칭하기로 한다) 어드레스를 사용하는 IPv4 네트워크(1530)와, IPv6 어드레스를 사용하는 IPv6 네트워크(1570)로 구성된다. 상기 Pv6 네트워크(1500)를 일 예로 하여 상기 UMTS 네트워크 구조를 설명하기로 한다.

<124> 먼저 사용자 단말기(UE: User Equipment, 이하 "UE"라 칭하기로 한다)(1511)는 UMTS 육상 무선 접속 네트워크(UTRAN: UMTS Terrestrial Radio Access Network, 이하 "UTRAN"이라 칭하기로 한다)(1513)와 접속되어 호(call)를 처리하며, 회선 서비스(CS: Circuit Service)와 패킷 서비스(PS: Packet Service)를 모두 지원한다. 특히, 본 발명에서 상기 UE(1511)는 IPv4 어드레스와 IPv6 어드레스를 모두 지원 가능한 듀얼 모드(dual mode) UE이다. 상기 UE(1511)는 상기 종래 기술 부분에서 설명한 바와 같이 트래픽 플로우 탬플릿(TFT: Traffic Flow Template, 이하 "TFT"라 칭하기로 한다) 정보를 구성하는데, 본 발명에서 상기 UE(1511)는 사용할 IP 어드레스를 그대로 사용하여 TFT 패킷 필터를 생성하거나 혹은 IP 어드레스의 일부분만을 사용하여 TFT 패킷 필터를 생성한다. 이렇게 IP 어드레스의 전부 혹은 일부를 사용하여 TFT 패킷 필터를 생성하는 과정은 하기에서 자세히 설명할 것이므로 여기서는 그 상세한 설명을 생략하기로 한다.

<125> 상기 UTRAN(1513)은 기지국(Node B)(도시하지 않음)과, 무선 네트워크 제어기(RNC: Radio Network Controller, 이하 "RNC"라 칭하기로 한다)(도시하지 않음)로 구성되며, 상기 기지국은 상기 UE(1511)과 Uu 인터페이스(interface)를 통해서 연결되며, 상기 RNC는 서비스 패킷 무선 서비스 지원 노드(SGSN: Serving GPRS Support Node, 이하 "SGSN"이라 칭하기로 한다)(1515)와 Iu 인터페이스를 통해서 연결된다. 여기서, 상기 패킷 무선 서비스(GPRS: General Packet Radio Service, 이하 "GPRS"라 칭하기로 한다)는 상기 UMTS 네트워크에서 수행하는 패킷 데이터 서비스이다. 상기 UTRAN(1513)은 상기 UE(1511)에서 에어(air)상으로 전송한 무선 데이터 혹은 제어 메시지(control message)들을 GPRS 터널링 프로토콜(GTP: GPRS Tunneling Protocol, 이하 "GTP"라 칭하기로 한다)을 사용하는 코어 네트워크(CN: Core Network)로 전달하기 위해 프로토콜 변환을 수행한다. 여기서, 상기 코어 네트워크는 상기 SGSN(1515)과 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node, 이하 "GGSN"이라 칭하기로 한다)(1519)를 통칭한다.

<126> 그리고, 상기 SGSN(1515)는 UE(1511)의 가입자 정보와, 위치 정보를 관리하는 네트워크 노드이다. 상기 SGSN(1515)는 상기 UTRAN(1513)과는 Iu 인터페이스를 통해 연결되며, GGSN(1519)과는 Gn 인터페이스를 통해 연결되어 데이터 및 제어 메시지 등을 송수신한다. 그리고 상기 SGSN(1515)는 홈위치 등록기(HLR: Home Location Register)(1517)와 Gr 인터페이스를 통해 연결되어 상기 가입자 정보 및 위치 정보를 관리한다.

<127> 상기 홈위치 등록기(1517)는 패킷 도메인(packet domain)의 가입자 정보 및 라우팅(routing) 정보 등을 저장한다. 상기 홈위치 등록기(1517)는 상기 SGSN(1515)과는 Gr 인터페이스를 통해 연결되며, 상기 GGSN(1519)과는 Gc 인터페이스를 통해 연결된다. 그리

고, 상기 홈위치 등록기(1517)는 UE(1511)의 로밍(roaming)등을 고려하여 다른 공중 육상 이동 통신 네트워크(PLMN: Public Land Mobile Network, 이하 "PLMN"이라 칭하기로 한다)에 위치할 수 있음은 물론이다. 그리고 상기 GGSN(1519)은 상기 UMTS 네트워크에 있어서 GTP의 종단이며, Gi 인터페이스를 통해 외부 네트워크와 연결되어 인터넷(internet), 혹은 패킷 도메인 네트워크(PDN: Packet Domain Network), 혹은 다른 PLMN 등과 연동할 수 있다. 상기 IPv6 네트워크(1500)는 상기 IPv4 네트워크(1550)와 제1보더 게이트웨이(Boader Gateway 1)(1520)를 통해 연결된다. 상기 제1보더 게이트웨이(1520)는 상기 IPv6 네트워크(1500)의 가장 종단에 위치하며, 메시지 필터링(message filtering), NAT(Network Address Translation) 등과 같은 기능을 수행한다.

<128> 특히, 본 발명에서 상기 제1보더 게이트웨이(1520)는 상기 IPv6 네트워크(1500)에서 전달받은 패킷 데이터를 제2보더 게이트웨이(1530)로 전달한다. 여기서, 상기 IPv6 네트워크(1500)에서 전달받은 패킷 데이터는 IPv6 어드레스를 가지지만, 상기 제2보더 게이트웨이(1530)가 전달된 IPv4 네트워크(1550)는 IPv4 어드레스만을 지원한다. 따라서, 상기 제1보더 게이트웨이(1520)는 상기 IPv6 네트워크(1500)에서 전달받은 패킷 데이터의 IPv6 어드레스의 하위 32비트를 추출하여 IPv4 헤더(header)를 생성하고, 상기 생성한 IPv4 헤더를 상기 패킷 데이터에 추가하여 IPv4 네트워크(1550)로 전달한다. 여기서, 상기 UMTS 통신 시스템은 상기 종래 기술 부분에서 설명한 바와 같이 IPv4 네트워크와 IPv6 네트워크간에 IP 통신이 가능하도록 IPv4 삽입 IPv6 어드레스(이하 "IPv4 embedded IPv6 어드레스"라 칭하기로 한다)를 사용한다. 여기서, 상기 IPv4 embedded IPv6 어드레스는 IPv4 호환(compatible) IPv6(이하 "IPv4 compatible IPv6"라 칭하기로 한다) 어드레스와, IPv4 매핑(mapped) IPv6(이하 "IPv4 mapped IPv6"라 칭하기로 한다)

어드레스로 정의된다. 한편, 상기 IPv4 네트워크(1550)는 상기 제2보더 게이트웨이(1530)에서 전달받은 패킷 데이터의 IPv4 헤더를 제거하고, 상기 IPv4 헤더가 제거된 패킷 데이터를 제3보더 게이트웨이(1540)를 통해 전달한다. 그러면 상기 제3보더 게이트웨이(1540)는 제4보더 게이트웨이(1560)를 통해 패킷 데이터를 전달하고, 결과적으로 IPv6 네트워크(1570)는 IPv6 어드레스를 가지는 패킷 데이터를 수신하게 된다. 상기 설명에서는 IPv6 네트워크(1500)에서 외부로 패킷 데이터가 유출되는 과정만을 설명하였지만, 이와는 반대로 외부에서 IPv6 네트워크(1500)로 패킷 데이터가 유입될 때도 그 IP 어드레스 버전에 따라 패킷데이터를 캡슐레이션(capsulation) 혹은 디캡슐레이션(de-capsulation)하여 처리하게 된다. 그리고, 하기의 설명에서 설명의 편의상 IPv4 어드레스를 가지는 패킷 데이터를 "IPv4 패킷 데이터"라 칭하기로 하며, IPv6 어드레스를 가지는 패킷 데이터를 "IPv6 패킷 데이터"라 칭하기로 한다.

<129> 또한, 상기 제2보더 게이트웨이(1530)는 IPv4 네트워크(1550)의 경계 라우터(router) 기능을 수행하며, 일반적인 IPv4 라우터 동작을 수행한다. 상기 제3보더 게이트웨이(1540)는 IPv4 네트워크(1550)의 경계 라우터 기능을 수행하며, 일반적인 IPv4 라우터 동작을 수행한다. 상기 제4보더 게이트웨이(1560)는 IPv6 네트워크(1570)의 경계 라우터 기능을 수행하며, 상기 제1보더 게이트웨이(1520)와 동일한 기능을 수행한다. IPv4/IPv6 서버(server)는 IPv4 패킷 데이터와 IPv6 패킷 데이터를 모두 수용할 수 있는 듀얼 모드의 서버로서, IPv4 네트워크(1550)를 경유하여 UMTS 네트워크의 UE(1511)와 통신을 하기 위해 IPv4-compatible IPv6 어드레스 혹은 IPv4-mapped IPv6 어드레스를 사용한다.

<130> 그러면 다음으로 도 16을 참조하여 본 발명의 실시예에서의 기능을 수행하기 위한 TFT 패킷 필터링 장치 내부 구조를 설명하기로 한다.

<131> 상기 도 16은 본 발명의 실시예에서의 기능을 수행하기 위한 TFT 패킷 필터링 장치 내부 구조를 도시한 블록도이다.

<132> 상기 도 16을 참조하면, 먼저 상기 TFT 패킷 필터링 장치는 크게 제어기(CPU: Central Processing Unit)(1600)와, 메모리(RAM: Random Access Memory)(1650)와, 분할 및 재조합기(SAR :Segmentation and Reassembly)(1670) 및 듀플렉서(Duplexer)(1690)로 구성된다. 상기 제어기(1600)는 GGSN의 Gi 인터페이스를 통해 외부 네트워크, 일 예로 인터넷(internet)으로부터 유입되는 패킷 데이터를 처리하며, 수학적 연산 및 스케쥴링(scheduling), 태스크(task) 관리 등과 같은 전반적인 제어 동작을 수행한다. 특히, 본 발명의 실시예에서 상기 제어기(1600)는 PSSB(Packet Service Slace Block) 태스크(1610)를 관리하며, 상기 도 16에 도시되어 있는 SIPC(S InterProcess Communications) 태스크는 해쉬 처리했으며 본 발명의 실시예와는 직접적인 관련이 없으므로 여기서는 그 상세한 설명을 생략하기로 한다. 여기서, 상기 PSSB 태스크(1010)는 GTP 터널(tunnel)을 통해 전달된 GTP-u 패킷 데이터나 외부 네트워크로, 일 예로 인터넷으로부터 수신된 IP 패킷 데이터를 수신하여 각종 프로토콜 처리를 한다.

<133> 그리고, 상기 PSSB 태스크(1010)는 TFT 패킷 필터링 프로시저(TFT Packet filtering Procedure)(1611)와, 패킷 프로세서(packet processor)(1613)로 구성된다. 상기 TFT 패킷 필터링 프로시저(1611)는 상기 TFT들에서 패킷 필터링을 수행하는 프로시저이며, 패킷 프로세서(1613)는 상기 TFT 패킷 필터링 프로시저(161)에서 TFT 패킷 필터링된 패킷을 처리한다. 상기 메모리(1650)는 TFT 테이블(TFT Table)(1651)과, 자원

테이블(resource table)(1653)을 구비한다. 상기 TFT 테이블(1651)은 상기 GGSN에 저장되어 있는 TFT들에 대한 정보들을 저장하고 있으며, 상기 TFT 패킷 필터링 프로시저(1611)는 상기 GGSN으로 유입되는 패킷 데이터들에 대해 상기 TFT 테이블(1651)을 참조하여 패킷 필터링을 수행한다. 여기서, 상기 TFT 테이블(1651)에 저장되어 있는 TFT 패킷 필터들은 본 발명에서 IPv4 compatible IPv6 어드레스와 IPv4 매핑(mapped) IPv6(이하 "IPv4 mapped IPv6"라 칭하기로 한다) 어드레스를 사용함에 따라 32비트의 IPv4 어드레스를 가지게 된다. 여기서, 상기 IPv4 compatible IPv6 어드레스는 상대방 네트워크가 IPv6 어드레스를 지원하며, 상대방, 즉 데스티네이션의 IPv4 어드레스를 알고 있고, IPv6 네트워크를 통해 통신하고자 할 경우 선택적으로 사용되는 어드레스이다. 또한, 상기 IPv4 mapped IPv6 어드레스는 상대방 네트워크가 IPv6 어드레스를 지원하지 않지만, IPv6 어드레스를 가지고 통신을 수행해야할 경우 선택적으로 사용되는 어드레스이다.

<134> 상기 분할 및 재조립기(1670)는 외부 네트워크로부터 입력되는 비동기 전송 모드(ATM: Asynchronous Transfer Mode) 셀(cell)들을 재조립(reassembly)하여 상기 PSSB 태스크(1610)내의 IN 경로로 전달하며, 상기 GGSN에서 외부 네트워크로 출력되는 패킷 데이터들, 즉 PSSB 태스크(1610)의 IN, P, S 등의 경로를 통해 전달된 패킷 데이터들을 ATM 셀 단위로 분할(segmentation)하여 상기 듀플렉서(1090)로 출력한다. 상기 듀플렉서(1090)는 상기 외부 네트워크로부터 입력되는 패킷 데이터들은 선택적으로 유입시키고, 상기 GGSN에서 출력되는 패킷 데이터들은 물리적으로(physical) 연결된 모든 블록(block)들로 전송한다.

<135> 또한 상기 도 16에서 설명한 TFT 패킷 필터링 장치는 유입되는 패킷 데이터에 대해서 TFT 패킷 필터링을 수행하기 위해서는 제2 PDP 컨텍스트 활성화 과정과, TFT 정보 저

장을 고려해야만 한다. 상기 TFT 패킷 필터링을 수행하기 위해 고려해야할 점들을 설명하기에 앞서 본 발명을 설명함에 있어 UMTS 네트워크 및 코어 네트워크(CN: Core Network) 구조는 상기 종래 기술 부분의 도 1 및 도 2에서 설명한 바와 동일한 구조를 가지며, 다만 TFT 패킷 필터링을 위한 부분만이 차별적인 구조를 가진다. 즉, 본 발명에서는 IPv4 삽입 IPv6 어드레스(이하 "IPv4 embedded IPv6 어드레스"라 칭하기로 한다)인 IPv4 compatible IPv6 어드레스와, IPv4 mapped IPv6 어드레스를 사용하는 경우를 가정하고 있으며, 따라서 TFT 패킷 필터링을 위한 TFT 패킷 필터를 상기 IPv4 embedded IPv6 어드레스인의 IPv4 어드레스만을 사용하여 TFT 패킷 필터링을 수행하기 때문이다. 또한 본 발명의 패킷 데이터 프로토콜(PDP: Packet Data Protocol, 이하 "PDP"라 칭하기로 한다) 컨텍스트(context), 즉 제1 PDP 컨텍스트(primary PDP context)와 제2 PDP 컨텍스트(second PDP context)의 활성화(activation) 과정은 상기 도 4 및 도 5에서 설명한 바와 동일한 과정을 거침에 유의하여야만 한다.

<136> 본 발명의 TFT 패킷 필터링을 수행하기 위해서는 첫 번째로, 상기에서 설명한 바와 같이 제2 PDP 컨텍스트 활성화 과정을 고려하여야한다.

<137> 상기 제2 PDP 컨텍스트 활성화 과정을 고려해야하는 이유는 상기 TFT가 제1 PDP 컨텍스트 활성화시에는 생성되지 않고 제2 PDP 컨텍스트 활성화 과정에서만 생성되기 때문이다. 상기 제2 PDP 컨텍스트 활성화 과정은 상기 도 5에서 설명한 바와 같이 UE(1511)가 SGSN(1515)으로 PDP 컨텍스트 활성화 요청(Activate Secondary PDP Context Request) 메시지를 전송함에 따라 상기 SGSN(1515)이 GGSN(1519)으로 PDP 컨텍스트 생성 요청(Create PDP Context Request) 메시지를 전송함에 따라 시작된다. 상기 도 5에서 설명한 바와 같이 TFT 정보는 UE(1511)에서 생성되며, 상기 PDP 컨텍스트 생성 요청 메시지에

포함되어 상기 GGSN(1519)에 전달된다. 그러면 상기 GGSN(1519)은 상기 PDP 컨텍스트 생성 요청 메시지에 포함되어 있는 TFT 정보를 가지고 제2 PDP 컨텍스트를 활성화시켜 제2 GTP 터널을 생성하고, 상기 생성된 제2 GTP 터널을 통해서 외부 네트워크로부터 유입되는 패킷 데이터들을 처리할 수 있게된다.

<138> 본 발명의 TFT 패킷 필터링을 수행하기 위해서는 두 번째로 TFT 정보 저장을 고려하여야 한다.

<139> 상기에서 설명한 바와 같이 UE(1511)으로부터 전달받은 TFT 정보는 상기 GGSN(1519)의 Gi 인터페이스에 저장되는데, 이때 상기 TFT 정보중 필요한 정보들, 즉 패킷 필터(packet filter)들의 개수, 패킷 필터 컨텐츠(packet filter contents) 등과 같은 정보들을 저장하여 외부 네트워크로부터 유입되는 패킷 데이터에 대한 TFT 패킷 필터링이 가능하도록 한다. 즉, 상기 TFT 정보는 상기 제2 PDP 컨텍스트 활성화 요구 메시지에 포함되어 상기 SGSN(1515)로 전달되고, PDP 컨텍스트 생성 요구 메시지에 포함되어 상기 GGSN(1519)으로 전달되는데, 상기 GGSN(1519)은 필요한 TFT 정보들만을 추출하여 저장하도록 한다.

<140> 본 발명은 상기 TFT 정보를 저장함에 있어서 두 가지 방법을 제안하며, 상기 두가지 방법들을 설명하면 다음과 같다.

<141> (1) IPv6 소스 어드레스 타입(이하 "IPv6 source address type"이라 칭하기로 한다) 방법

<142> 상기에서 설명한 바와 같이 UE(1511)에서 생성한 TFT 정보는 GGSN(1519)에 저장되는데, 상기 GGSN(1519)은 상기 UE(1511)에서 전달한 정보중 필요한 정보만을 추출하여

TFT 정보로 저장한다. 즉, 상기 GGSN(1519)은 패킷 필터의 개수, 패킷 필터 등으로 구성하여 패킷 필터링이 용이하도록 TFT 정보를 저장한다. 이때 TFT 패킷 필터의 종류가 IPv6 source address type이고 해당 필터 계수가 IPv4 embedded IPv6 어드레스일 경우 상기 GGSN(1519)은 TFT 정보로 상기 IPv4 embedded IPv6 어드레스를 나타내기 위한 128비트의 어드레스값과 128비트의 마스크(mask)값을 저장하지 않고 하위 32비트, 즉 IPv4 embedded IPv6의 IPv4 어드레스를 나타내는 하위 32비트들만을 선택하여, 32비트의 어드레스값과 32비트의 마스크값만을 저장한다. 즉, 상기 TFT 패킷 필터는 실제 IPv6 source address type을 가지지만 상기 TFT 패킷 필터에 저장되는 필터 계수는 IPv4 어드레스 포맷을 가지게 된다.

<143> 상기 GGSN(1519)은 상기 UE(1511)에서 전송하는 제2PDP 컨텍스트 활성화 요청 메시지에 포함되어 있는 TFT 정보 중 필요한 정보만을 이용하여 TFT 정보를 저장하는데, 상기 GGSN(1519)에 저장되는, 즉 상기 TFT 패킷 필터링 장치의 메모리(1650)에 저장되는 TFT 정보를 도 17을 참조하여 설명하기로 한다.

<144> 상기 도 17은 도 16의 TFT 테이블(1651)에 저장되는 TFT 정보를 도시한 도면이다.

<145> 상기 도 17을 참조하면, 패킷 필터 넘버(Number of Packet filters) 영역(1711)과, 패킷 필터 ID(packet filter identifier) 영역들(1713, 1723, 1733, 1743, 1753)과, 패킷 필터 평가 순위(packet filter evaluation precedence) 영역(도시하지 않음)과, 패킷 필터 컨텐츠(packet filter contents) 영역들(1715,1725,1735,1745,1755)로 구분된다. 상기 패킷 필터 넘버 영역(711)은 해당 TFT에 저장되는 패킷 필터들의 수를 나타내며, 상기 패킷 필터 ID 영역들(1713, 1723, 1733, 1743, 1753)은 상기 TFT에 저장되어 있는 다수개의 패킷 필터들 각각을 구분하기 위한 패킷 필터 ID를 나타낸다. 그리고, 상기 패

킷 필터 ID 각각에 상응하여 패킷 필터 평가 순위 영역 및 패킷 필터 컨텐츠 영역들(1715,1725,1735,1745,1755) 각각이 저장된다. 한편, 상기 도 17에 저장되는 TFT 정보는 일반적인 TFT 정보, 즉 도 6에 도시되어 있는 TFT 정보들 중 본 발명의 TFT 패킷 필터링에 필요한 정보들만을 별도로 선택한 것이다. 본 발명에서는 IPv4 embedded IPv6 어드레스의 TFT 패킷 필터링을 수행하므로 소스 어드레스(source address) 및 데스티네이션 어드레스(destination address) 컨텐츠를 중요하게 고려한다.

<146> 일 예로, UE(1511)로부터 수신한 TFT 정보의 첫 번째 패킷 필터 컨텐츠가 IPv4-compatible IPv6 주소 ::3.2.2.1이고 프로토콜이 UDP일 경우, 상기 GGSN(1519)은 IPv6 source address type 방법을 사용하여 IPv6 source address type 3.2.2.1과 protocol type UDP의 컨텐츠를 가지는 패킷 필터를 생성하여 상기 TFT 패킷 필터링 장치 메모리(1650)의 TFT 테이블(1651)에 저장하는 것이다.

<147> 상기에서는 IPv6 source address type 방법을 사용하여 TFT 정보를 저장하는 경우를 설명하였으며, 다음으로 IPv4 삽입 IPv6 소스 어드레스 타입(이하 "IPv4 Embedded IPv6 source address type"라 칭하기로 한다) 방법을 사용하여 TFT 정보를 저장하는 경우를 설명하기로 한다.

<148> (2) IPv4 Embedded IPv6 source address type 방법

<149> 상기에서 설명한 바와 같이 UE(1511)가 TFT 정보를 생성하는데, IP 어드레스가 IPv4 Embedded IPv6 source address일 경우 상기 UE(1511)는 TFT 패킷 필터 타입을 IPv4 Embedded IPv6 source address type으로 설정하고, IPv6 어드레스의 하위 32비트만을 추출한다. 상기 UE(1511)는 상기 추출한 IPv4 Embedded IPv6 source address의 하위 32비트를 가지고 새로운 TFT 패킷 필터를 구성해서 GGSN(1519)으로 전송한다. 이렇게

UE(1511)가 IPv4 Embedded IPv6 source address의 하위 32비트만을 추출하여 새로운 TFT 패킷 필터를 구성하여 송신하는 방법이 상기 IPv4 Embedded IPv6 source address type 방법이다. 상기 IPv4 Embedded IPv6 source address type 방법을 지원하기 위해서는 상기 종래 기술 부분의 표 2에서 설명한 패킷 필터 컴퍼넌트 타입들에 IPv4 Embedded IPv6 source address type을 추가해야만 한다. 상기 IPv4 Embedded IPv6 source address type의 패킷 필터 컴퍼넌트 타입 ID는 "0010 0001"로 설정하기로 한다. 여기서, 상기 "0010 0001"은 상기 패킷 필터 컴퍼넌트 타입 ID들중 예약(reserved)되어 있던 값이다.

<150> 결과적으로, 상기에서 설명한 바와 같이 IPv6 source address type 방법을 사용할 경우에는 TFT 패킷 필터가 IPv6 source address type이며, 따라서 저장된 TFT 패킷 필터의 길이가 32비트가 되고, 이와는 달리 상기 IPv4 Embedded IPv6 source address type 방법을 사용할 경우에는 TFT 패킷 필터가 IPv4 Embedded IPv6 source address type이 되며, 저장된 TFT 패킷 필터의 길이는 동일하다.

<151> 다음으로 도 18a 및 도 18b를 참조하여 상기 IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링 과정을 설명하기로 한다.

<152> 상기 도 18a 내지 도 18b는 IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링 과정을 도시한 순서도이다.

<153> 상기 도 18a를 참조하면, 먼저 1811단계에서 GGSN(1519)은 Gi 인터페이스를 통해 IP 패킷 데이터를 전달받으면 1813단계로 진행한다. 상기 1813단계에서 GGSN(1519)은 상기 전달받은 IP 패킷 데이터의 데스티네이션 어드레스를 확인하여 PDP 어드레스와 매칭되는 정보에 제2호(secondary call)가 설정되어 있는지 검사한다. 여기서, 상기 제2호가 설정되어 있는지 검사하는 이유는 제2GTP 터널이 존재하는지를 검사하기 위해서이다.

즉, 상기 제2GTP 터널이 존재하지 않을 경우에는 TFT 패킷 필터링이 불가능하기 때문에 상기 제2호가 존재하는지를 검사하는 것이다. 상기 검사 결과 상기 제2호가 설정되어 있지 않을 경우 상기 GGSN(1519)은 1827단계로 진행한다. 상기 1827단계에서 상기 GGSN(1519)은 제1GTP 터널을 선택하고 1821단계로 진행한다.

<154> 한편, 상기 1813단계에서 검사 결과 상기 제2호가 설정되어 있을 경우 상기 GGSN(1519)은 1815단계로 진행한다. 상기 1815단계에서 상기 GGSN(1519)은 제2GTP 터널을 선택하여 첫 번째 TFT 정보부터 우선 순위가 가장 높은 TFT 패킷 필터를 선택하고 1851단계로 진행한다. 상기 1851단계에서 상기 GGSN(1519)은 상기 최우선 순위 TFT 패킷 필터가 IPv6 source address type인지를 검사한다. 상기 검사 결과 상기 최우선 순위 TFT 패킷 필터가 IPv6 source address type이 아닐 경우 상기 GGSN(1519)은 1867단계로 진행한다. 상기 1867단계에서 상기 GGSN(1519)은 일반적인 TFT 패킷 필터링 과정을 수행하고 1869단계로 진행한다. 상기 1851단계에서 검사 결과 상기 최우선 순위 TFT 패킷 필터가 IPv6 source address type일 경우 상기 GGSN(1519)은 1853단계로 진행한다. 상기 1853단계에서 상기 GGSN(1519)은 상기 Gi 인터페이스를 통해 전달받은 IP 패킷 데이터의 버전, 소스 어드레스의 IP 버전이 IPv6인지 검사한다. 상기 검사 결과 상기 전달받은 IP 패킷 데이터의 버전이 IPv6가 아닐 경우 상기 GGSN(1519)은 1855단계로 진행한다. 상기 1855단계에서 상기 GGSN(1519)은 상기 첫 번째 TFT 정보에 다른 TFT 패킷 필터가 존재하는지 검사한다. 상기 검사 결과 다른 TFT 패킷 필터가 존재할 경우 상기 GGSN(1519)은 1857단계로 진행한다. 상기 1857단계에서 상기 GGSN(1519)은 상기 다른 TFT 패킷 필터들중 우선 순위가 가장 높은 TFT 패킷 필터를 선택한 후 상기 1851단계로 되돌아간다. 또한, 상기 1855단계에서 검사 결과 다른 TFT 패킷 필터가 존재하지 않을 경우 상기

GGSN(1519)은 1825단계로 진행한다. 상기 1825단계에서 상기 GGSN(1519)은 다음 TFT 정보가 존재하는지 검사한다. 상기 검사 결과 다음 TFT 정보가 존재할 경우 상기 GGSN(1519)은 1823단계로 진행한다. 상기 1823단계에서 상기 GGSN(1519)은 다음 TFT 정보를 선택하고 상기 1815단계로 되돌아간다. 또한, 상기 1825단계에서 검사 결과 다음 TFT 정보가 존재하지 않을 경우 상기 GGSN(1519)은 1827단계로 진행한다. 상기 1827단계에서 상기 GGSN(1519)은 제1GTP 터널을 선택하고 1817단계로 진행한다.

<155> 한편, 상기 1853단계에서 검사 결과 상기 전달받은 IP 패킷 데이터의 버전이 IPv6일 경우 상기 GGSN(1519)은 1859단계로 진행한다. 상기 1859단계에서 상기 GGSN(1519)은 TFT 패킷 필터의 길이가 32비트인지 검사한다. 상기 검사 결과 상기 TFT 패킷 필터의 길이가 32비트가 아닐 경우 상기 GGSN(1519)은 상기 1867단계로 진행한다. 여기서, 상기 TFT 패킷 필터의 길이가 32비트가 아니라는 것은 일반적인 128비트의 IPv6 어드레스를 의미하는 것이므로 상기 1867단계로 진행하여 일반적인 TFT 패킷 필터링을 수행하는 것이다. 상기 1859단계에서 검사 결과 상기 TFT 패킷 필터의 길이가 32비트일 경우 상기 GGSN(1519)은 1861단계로 진행한다. 상기 1861단계에서 상기 GGSN(1519)은 상기 전달받은 IP 패킷 데이터의 소스 어드레스가 IPv4 Embedded IPv6 어드레스인지 검사한다. 상기 검사 결과 상기 소스 어드레스가 IPv4 Embedded IPv6 어드레스가 아닐 경우 상기 GGSN(1519)은 상기 1867단계로 진행한다. 여기서, 상기 소스 어드레스가 IPv4 Embedded IPv6 어드레스가 아니라는 것은 일반적인 32비트 IPv4 어드레스임을 나타내므로 상기 1867단계에서 일반적인 TFT 패킷 필터링을 수행하는 것이다.

<156> 상기 1861단계에서 검사 결과 상기 소스 어드레스가 IPv4 Embedded IPv6 어드레스일 경우 상기 GGSN(1519)은 1863단계로 진행한다. 상기 1863단계에서 상기

GGSN(1519)은 상기 소스 어드레스의 하위 32비트를 추출하고 1865단계로 진행한다. 상기 1865단계에서 상기 GGSN(1519)은 상기 추출한 32비트를 가지고 TFT 패킷 필터링을 수행하고 1869단계로 진행한다. 여기서, 상기 1865단계에서 수행하는 TFT 패킷 필터링은 상기에서 제안한 IPv6 source address type 방법을 사용하는 것이다. 상기 1869단계에서 상기 GGSN(1519)은 상기 TFT 패킷 필터링이 성공했는지를 검사한다. 상기 검사 결과 상기 TFT 패킷 필터링이 성공하지 않았을 경우 상기 GGSN(1519)은 1855단계로 진행한다. 상기 1869단계에서 검사 결과 상기 TFT 패킷 필터링이 성공했을 경우 상기 GGSN(1519)은 1817단계로 진행한다.

<157> 상기 1817단계에서 상기 GGSN(1519)은 현재 TFT 정보와 대응되는 GTP 터널을 선택하고 1821단계로 진행한다. 상기 1821단계에서 상기 GGSN(1519)은 상기 전달받은 IP 패킷 데이터를 처리하기 위한 패킷 프로시져(packet procedure)를 수행하고 종료한다.

<158> 상기 도 18a 및 도 18b에서는 상기 IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 과정을 설명하였으며, 다음으로 도 19a 및 도 19b를 참조하여 상기 IPv4 Embedded IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 과정을 설명하기로 한다.

<159> 상기 도 19a 내지 도 19b는 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링 과정을 도시한 순서도이다.

<160> 상기 도 19a를 참조하면, 먼저 1911단계에서 GGSN(1519)은 Gi 인터페이스를 통해 IP 패킷 데이터를 전달받으면 1913단계로 진행한다. 상기 1913단계에서 GGSN(1519)은 상기 전달받은 IP 패킷 데이터의 데스티네이션 어드레스를 확인하여 PDP 어드레스와 매칭되는 정보에 제2호(secondary call)가 설정되어 있는지 검사한다. 여기서, 상기 제2호가

설정되어 있는지 검사하는 이유는 상기에서 설명한 바와 같이 제2GTP 터널이 존재하는지를 검사하기 위해서이다. 즉, 상기 제2GTP 터널이 존재하지 않을 경우에는 TFT 패킷 필터링이 불가능하기 때문에 상기 제2호가 존재하는지를 검사하는 것이다. 상기 검사 결과 상기 제2호가 설정되어 있지 않을 경우 상기 GGSN(1519)은 1927단계로 진행한다. 상기 1927단계에서 상기 GGSN(1519)은 제1GTP 터널을 선택하고 1917단계로 진행한다.

<161> 한편, 상기 1913단계에서 검사 결과 상기 제2호가 설정되어 있을 경우 상기 GGSN(1519)은 1915단계로 진행한다. 상기 1915단계에서 상기 GGSN(1519)은 제2GTP 터널을 선택하여 첫 번째 TFT 정보부터 우선 순위가 가장 높은 TFT 패킷 필터를 선택하고 1951단계로 진행한다. 상기 1951단계에서 상기 GGSN(1519)은 상기 최우선 순위 TFT 패킷 필터가 IPv4 Embedded IPv6 address 타입인지를 검사한다. 상기 검사 결과 상기 최우선 순위 TFT 패킷 필터가 IPv4 Embedded IPv6 address 타입이 아닐 경우 상기 GGSN(1519)은 1953단계로 진행한다. 상기 1953단계에서 상기 GGSN(1519)은 일반적인 TFT 패킷 필터링 과정을 수행하고 1965단계로 진행한다. 상기 1951단계에서 검사 결과 상기 최우선 순위 TFT 패킷 필터가 IPv4 Embedded IPv6 address 타입일 경우 상기 GGSN(1519)은 1955단계로 진행한다. 상기 1955단계에서 상기 GGSN(1519)은 상기 전달받은 IP 패킷 데이터의 소스 어드레스가 IPv4 Embedded IPv6 어드레스인지 검사한다. 상기 검사 결과 상기 전달받은 IP 패킷 데이터의 소스 어드레스가 IPv4 Embedded IPv6 어드레스가 아닐 경우 상기 GGSN(1519)은 1957단계로 진행한다. 상기 1957단계에서 상기 GGSN(1519)은 상기 첫 번째 TFT 정보에 다른 TFT 패킷 필터가 존재하는지 검사한다. 상기 검사 결과 다른 TFT 패킷 필터가 존재할 경우 상기 GGSN(1519)은 1959단계로 진행한다. 상기 1959단계에서 상기 GGSN(1519)은 상기 다른 TFT 패킷 필터들중 우선 순위가 가장 높은 TFT 패킷 필터를 선

택한 후 상기 1951단계로 되돌아간다. 또한, 상기 1957단계에서 검사 결과 다른 TFT 패킷 필터가 존재하지 않을 경우 상기 GGSN(1519)은 1925단계로 진행한다. 상기 1925단계에서 상기 GGSN(1519)은 다음 TFT 정보가 존재하는지 검사한다. 상기 검사 결과 다음 TFT 정보가 존재할 경우 상기 GGSN(1519)은 1923단계로 진행한다. 상기 1923단계에서 상기 GGSN(1519)은 다음 TFT 정보를 선택하고 상기 1915단계로 되돌아간다. 또한, 상기 1925단계에서 검사 결과 다음 TFT 정보가 존재하지 않을 경우 상기 GGSN(1519)은 1927단계로 진행한다. 상기 1927단계에서 상기 GGSN(1519)은 제1GTP 터널을 선택하고 1921단계로 진행한다.

<162> 한편, 상기 1955단계에서 검사 결과 상기 전달받은 IP 패킷 데이터의 소스 어드레스가 IPv4 Embedded IPv6 어드레스일 경우 상기 GGSN(1519)은 1961단계로 진행한다. 상기 1961단계에서 상기 GGSN(1519)은 상기 IPv4 Embedded IPv6 어드레스의 하위 32비트를 추출한 후 1963단계로 진행한다. 상기 1963단계에서 상기 GGSN(1519)은 상기 추출한 32비트를 가지고 TFT 패킷 필터링을 수행한 후 1965단계로 진행한다. 상기 1965단계에서 상기 GGSN(1519)은 상기 TFT 패킷 필터링이 성공했는지를 검사한다. 상기 검사 결과 상기 TFT 패킷 필터링이 성공하지 않았을 경우 상기 GGSN(1519)은 상기 1957단계로 진행한다. 상기 1965단계에서 검사 결과 상기 TFT 패킷 필터링이 성공했을 경우 상기 GGSN(1519)은 1917단계로 진행한다. 상기 1917단계에서 상기 GGSN(1519)은 현재 TFT 정보와 대응되는 GTP 터널을 선택하고 1921단계로 진행한다. 상기 1921단계에서 상기 GGSN(1519)은 상기 전달받은 IP 패킷 데이터를 처리하기 위한 패킷 프로시져(packet procedure)를 수행하고 종료한다.

<163> 다음으로 도 20을 참조하여 일반적인 TFT 패킷 필터링을 설명하기로 한다.

<164> 상기 도 20은 도 16의 TFT 패킷 필터링 프로시저(1611)의 일반적인 TFT 패킷 필터링 동작을 개략적으로 도시한 도면이다.

<165> 상기 도 20을 참조하면, 먼저 외부 네트워크로부터 IP 패킷 데이터(2000)가 GGSN(1519)의 Gi 인터페이스를 통해 입력되면, 즉 듀플렉서(1690)를 통해 IP 패킷 데이터(2000)가 입력되면, 상기 입력된 IP 패킷 데이터(2000)를 분할 및 재조립기(1670)를 통해 TFT 패킷 필터링 프로시저(1611)로 전달된다. 상기 TFT 패킷 필터링 프로시저(1611)는 메모리(1650)의 TFT 테이블(1651)에 저장되어 있는 TFT 정보들을 가지고서 TFT 패킷 필터링을 수행한다. 상기 TFT 테이블(1651)에 저장되어 있는 TFT 정보가 상기 도 20에 도시되어 있는 바와 같이 TFT 1과 TFT 2의 두 개의 TFT 정보일 경우, 상기 TFT 패킷 필터링 프로시저(1651)는 먼저 상기 IP 패킷 데이터(2000)를 TFT 1의 패킷 필터 1부터 TFT 패킷 필터링 시도한다. 여기서, 상기 IP 패킷 데이터(2000)를 살펴보면, 서비스 타입(TOS: Type Of Service)이 0x1F이며, 프로토콜은 TCP(6)이며, 소스 어드레스(source address)는 2.2.2.2이며, 데스티네이션 어드레스(destination address)는 3.3.3.3이며, 소스 포트(source port)는 5000이며, 데스티네이션 포트(destination port)는 50이다.

<166> 그러면 상기 IP 패킷 데이터(2000)를 TFT 1의 패킷 필터 1에 TFT 패킷 필터링 시도하면 상기 TFT 1의 패킷 필터 1의 소스 어드레스는 1.1.1.1이기 때문에 매핑되지 않아 TFT 패킷 필터링이 실패하게 된다. 그러면 상기 TFT 패킷 필터링 프로시저(1611)는 상기 IP 패킷 데이터(2000)에 대해 상기 TFT 1의 패킷 필터 2로 패킷 필터링을 시도한다. 그러나, 상기 TFT 1의 패킷 필터 2는 그 패킷 필터 컨텐츠가 소스 포트 범위 100~1000이므로 상기 IP 패킷 데이터(2000)의 소스 포트 5000에 매핑되지 않아 역시 TFT 패킷 필터링에 실패한다. 이런식으로 상기 입력된 IP 패킷 데이터(2000)에 매핑되는 TFT 패킷 필터

를 검색하게 되고, 상기 IP 패킷 데이터(2000)와 매칭되는 TFT 패킷 필터를 통해 필터링하고 해당 GTP 터널을 통해서 상기 IP 패킷 데이터(2000)를 SGSN(1515)로 전달한다. 상기 도 20에서는 IP 패킷 데이터(2000)의 데스티네이션 포트와 TFT 2의 패킷 필터 5의 데스티네이션 포트 범위가 매칭되므로 상기 IP 패킷 데이터(2000)는 상기 TFT 2에 해당하는 GTP 터널을 사용하게 된다. 물론 외부 네트워크로부터 유입된 패킷 데이터에 대한 TFT 패킷 필터링 과정 자체는 상기 종래 기술 부분의 도 10에서 설명한 방식과 동일하다.

<167> 다음으로 도 21을 참조하여 상기 IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링을 설명하기로 한다.

<168> 상기 도 21은 도 16의 TFT 패킷 필터링 프로시저(1611)가 IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 동작을 개략적으로 도시한 도면이다.

<169> 상기 도 21을 참조하면, 먼저 외부 네트워크로부터 IP 패킷 데이터(2100)가 GGSN(1519)의 Gi 인터페이스를 통해 입력되면, 즉 듀플렉서(1690)를 통해 IP 패킷 데이터(2100)가 입력되면, 상기 입력된 IP 패킷 데이터(2100)를 분할 및 재조립기(1670)를 통해 TFT 패킷 필터링 프로시저(1611)로 전달된다. 상기 TFT 패킷 필터링 프로시저(1611)는 메모리(1650)의 TFT 테이블(1651)에 저장되어 있는 TFT 정보들을 가지고서 TFT 패킷 필터링을 수행한다. 상기 TFT 테이블(1651)에 저장되어 있는 TFT 정보가 상기 도 21에 도시되어 있는 바와 같이 TFT 1과 TFT 2의 두 개의 TFT 정보일 경우, 상기 TFT 패킷 필터링 프로시저(1651)는 먼저 상기 IP 패킷 데이터(2100)를 TFT 1의 패킷 필터 1부터 TFT 패킷 필터링 시도한다. 여기서, 상기 IP 패킷 데이터(2100)를 살펴보면, 서비스 타입이 0x1F이며, 프로토콜은 TCP(6)이며, 소스 어드레스는 ::10.3.8.112이며, 데스티네

이션 어드레스는 ::10.2.3.54이며, 소스 포트는 5000이며, 데스티네이션 포트는 252다. 여기서, 상기 소스 어드레스와 데스티네이션 어드레스는 IPv4 compatible IPv6 어드레스로서, 하위 32비트만이 표시된 것이다.

<170> 그러면 상기 IP 패킷 데이터(2100)를 TFT 1의 패킷 필터 1에 TFT 패킷 필터링 시도하면 상기 TFT 1의 패킷 필터 1의 소스 어드레스는 10.3.8.112이기 때문에 TFT 패킷 필터링에 성공한다. 그러면 상기 TFT 패킷 필터링 프로시저(1611)는 IP 패킷 데이터(2100)와 매칭되는 TFT 패킷 필터를 통해 필터링하고 해당 GTP 터널을 통해서 상기 패킷 데이터(2100)를 SGSN(1515)로 전달한다. 상기 도 21에서는 패킷 데이터(2100)의 소스 어드레스와 상기 TFT 1의 패킷 필터 1의 소스 어드레스가 매칭되므로 상기 IP 패킷 데이터(2100)는 상기 TFT 1에 해당하는 GTP 터널을 사용하게 된다.

<171> 다음으로 도 22를 참조하여 상기 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링을 설명하기로 한다.

<172> 상기 도 22는 도 16의 TFT 패킷 필터링 프로시저(1611)가 IPv4 Embedded IPv6 source address type 방법을 사용하여 TFT 패킷 필터링하는 동작을 개략적으로 도시한 도면이다.

<173> 상기 도 22를 참조하면, 먼저 외부 네트워크로부터 IP 패킷 데이터(2200)가 GGSN(1519)의 Gi 인터페이스를 통해 입력되면, 즉 듀플렉서(1690)를 통해 IP 패킷 데이터(2200)가 입력되면, 상기 입력된 IP 패킷 데이터(2200)를 분할 및 재조립기(1670)를 통해 TFT 패킷 필터링 프로시저(1611)로 전달된다. 상기 TFT 패킷 필터링 프로시저(1611)는 메모리(1650)의 TFT 테이블(1651)에 저장되어 있는 TFT 정보들을 가지고서 TFT 패킷 필터링을 수행한다. 상기 TFT 테이블(1651)에 저장되어 있는 TFT 정보가 상기 도

22에 도시되어 있는 바와 같이 TFT 1과 TFT 2의 두 개의 TFT 정보일 경우, 상기 TFT 패킷 필터링 프로시저(1651)는 먼저 상기 IP 패킷 데이터(2200)를 TFT 1의 패킷 필터 1부터 TFT 패킷 필터링 시도한다. 여기서, 상기 IP 패킷 데이터(2200)를 살펴보면, 서비스 타입이 0x1F이며, 프로토콜은 TCP(6)이며, 소스 어드레스는 ::FFFF:10.3.2.1이며, 데스티네이션 어드레스는 ::FFFF:10.2.3.54이며, 소스 포트는 5000이며, 데스티네이션 포트는 50다. 여기서, 상기 소스 어드레스와 데스티네이션 어드레스는 IPv4 mapped IPv6 어드레스로서, 하위 32비트만이 표시된 것이다.

<174> 첫 번째로, 상기 TFT 패킷 필터링 프로시저(1611)는 상기 IP 패킷 데이터(2200)를 TFT 1의 패킷 필터 1부터 TFT 패킷 필터링 시도하면 상기 TFT 1의 패킷 필터 1의 소스 어드레스는 2002::AF10:E9이기 때문에 TFT 패킷 필터링에 실패하며, 상기 TFT 1의 패킷 필터 2의 소스 포트 범위는 100~1000이기 때문에 TFT 패킷 필터링에 실패하며, 상기 TFT 1의 패킷 필터 3의 프로토콜은 ICMP(1)이기 때문에 TFT 패킷 필터링에 실패한다. 두 번째로, 상기 TFT 패킷 필터링 프로시저(1611)는 TFT 2의 패킷 필터 1로 TFT 패킷 필터링 시도하면 IPv4 Embedded type 1이 10.3.2.1이기 때문에 TFT 패킷 필터링에 성공한다. 그러면 상기 TFT 패킷 필터링 프로시저(1611)는 IP 패킷 데이터(2200)와 매칭되는 TFT 패킷 필터를 통해 필터링하고 해당 GTP 터널을 통해서 상기 패킷 데이터(2200)를 SGSN(1515)로 전달한다. 상기 도 22에서는 패킷 데이터(2200)의 소스 어드레스와 상기 TFT 2의 패킷 필터 1의 IPv4 Embedded type 1이 매칭되므로 상기 IP 패킷 데이터(2200)는 상기 TFT 2에 해당하는 GTP 터널을 사용하게 된다.

<175> 다음으로 도 23을 참조하여 본 발명의 IPv6 source address type 방법 및 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링에 따른 비트 연산량과 일반적인 TFT 패킷 필터링에 따른 비트 연산량을 비교하기로 한다.

<176> 상기 도 23은 본 발명의 IPv6 source address type 방법 및 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터링에 따른 비트 연산량과 일반적인 TFT 패킷 필터링에 따른 비트 연산량을 비교적으로 도시한 도면이다.

<177> 상기 도 23을 참조하면, 먼저 TFT 패킷 필터링 횟수에 따라 IPv6 어드레스의 128 비트를 그대로 사용할 경우와 IPv6 어드레스의 128비트중 32비트를 추출하여 사용할 경우의 비트 연산량이 도시되어 있다. 즉, TFT 패킷 필터링 횟수가 1000회일 경우와, 10000회일 경우와, 100000회일 경우와, 1000000회일 경우의 128 비트 연산량과 32비트 연산량이 각각 도시되어 있다. 상기 도 23에 도시되어 있는 바와 같이 128비트를 그대로 사용할 경우와 32비트를 사용할 경우는 그 연산량에 있어서 큰 차이가 발생하게 된다.

<178> 한편, 상기에서 설명한 바와 같이 IPv4 Embedded IPv6 source address type 방법은 UE(1511)가 TFT 패킷 필터 타입을 IPv4 Embedded IPv6 source address type으로 설정하고, IPv6 어드레스의 하위 32비트만을 추출한 후, 상기 추출한 IPv4 Embedded IPv6 source address의 하위 32비트를 가지고 새로운 TFT 패킷 필터를 구성한다. 즉, 상기 IPv4 Embedded IPv6 source address type 방법은 상기 IPv6 source address type 방법과는 달리 UE(1511)가 TFT를 구성하는 방법이 상이하다. 이를 도 24를 참조하여 설명하기로 한다.

<179> 상기 도 24는 IPv4 Embedded IPv6 source address type 방법을 사용할 경우의 TFT 패킷 필터 생성 과정을 도시한 순서도이다.

<180> 상기 도 24를 참조하면, 먼저 2411단계에서 상기 UE(1511)는 임의의 변수 i를 0으로 설정하고(i = 0), 임의의 변수 Max_filter를 x로 설정하고 2413단계로 진행한다. 여기서, 상기 x는 하나의 TFT 내에 구성할 수 있는 패킷 필터들의 개수를 나타나며, 상기에서 설명한 바와 같이 현재 TFT 내에는 일 예로 8개까지의 패킷 필터들을 구성할 수 있기 때문에, 상기 x는 1~8까지의 정수들 중 한 정수값을 가진다. 상기 하나의 TFT 내에 구성할 수 있는 패킷 필터들의 개수 x는 상기 UE(1511)의 어플리케이션(application)에 의해 결정된다. 상기 2413단계에서 상기 UE(1511)는 상기 변수 i의 값이 상기 변수 Max_filter 값 미만인지를 검사한다. 상기 검사 결과 변수 i의 값이 상기 변수 Max_filter 값 이상일 경우 상기 UE(1511)는 현재까지의 과정을 종료하고, 만약 상기 검사 결과 상기 변수 i의 값이 상기 변수 Max_filter 값 미만일 경우에는 2415단계로 진행한다. 상기 2415단계에서 상기 UE(1511)는 상기 TFT 패킷 필터를 구성할 IP 어드레스가 IPv4 Embedded IPv6 source address type인지를 검사한다. 상기 검사 결과 상기 TFT 패킷 필터를 구성할 IP 어드레스가 IPv4 Embedded IPv6 source address type이 아닐 경우 상기 UE(1511)는 2417단계로 진행한다. 상기 2417단계에서 상기 UE(1511)는 일반적인 TFT 패킷 필터 생성 방법과 동일한 방법으로 TFT 패킷 필터를 구성하고 2423단계로 진행한다. 한편, 상기 검사 결과 상기 TFT 패킷 필터를 구성할 IP 어드레스가 IPv4 Embedded IPv6 source address type일 경우 상기 UE(1511)는 2419단계로 진행한다.

<181> 상기 2419단계에서 상기 UE(1511)는 상기 생성할 패킷 필터 타입을 IPv4 Embedded IPv6 source address type으로 설정한 후 2421단계로 진행한다. 상기 2421단계에서 상기 UE(1511)는 상기 IPv4 Embedded IPv6 address의 하위 32비트를 추출한 후 2423단계로 진행한다. 상기 2423단계에서 상기 UE(1511)는 상기 추출한 하위 32비트를 가지고 패킷

필터를 생성하고, 상기 생성한 패킷 필터를 TFT에 저장하고 2425단계로 진행한다. 상기 2425단계에서 상기 UE(1511)는 상기 변수 i의 값을 1증가시킨 후(i = i + 1) 상기 2413 단계로 진행한다.

<182> 한편 본 발명의 상세한 설명에서는 구체적인 실시예에 관해 설명하였으나, 본 발명의 범위에서 벗어나지 않는 한도내에서 여러 가지 변형이 가능함은 물론이다. 그러므로 본 발명의 범위는 설명된 실시예에 국한되어 정해져서는 안되며 후술하는 특허청구의 범위뿐만 아니라 이 특허청구의 범위와 균등한 것들에 의해 정해져야 한다.

【발명의 효과】

<183> 상술한 바와 같은 본 발명은, 이동통신시스템에서 외부 네트워크로부터 유입되는 패킷 데이터들의 IP 어드레스의 타입이 IPv4 embedded IPv6 어드레스일 경우 128비트를 그대로 사용하는 것이 아니라 하위 32비트만을 사용함으로써 TFT 패킷 필터링시 비트 연산량을 최소화한다는 이점을 가진다. 즉, 한번의 TFT 패킷 필터링마다 96비트에 대한 연산량이 감소하기 때문에 비트 연산량이 최소화되는 것이다.

<184> 또한, 본 발명은 IP 어드레스의 타입이 IPv4 embedded IPv6 어드레스일 경우 TFT 패킷 필터를 구성할 때 128비트가 아닌 32비트를 이용하기 때문에 TFT 패킷 필터의 저장 용량을 최소화시킨다는 이점을 가진다. 그래서, 이동 통신 시스템 전체의 자원 효율성을 증가시키게 된다는 이점을 가진다.

【특허청구범위】

【청구항 1】

제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서,

소스 IP 어드레스의 버전에 상응하여 상기 소스 IP 어드레스로부터 상기 IP 버전에 따른 정보를 추출하는 과정과,

상기 추출한 정보를 포함하는 TFT 정보를 생성하여 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 2】

제1항에 있어서,

상기 소스 IP 어드레스의 버번에 상응하여 상기 정보를 추출하는 과정은 상기 소스 IP 어드레스가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 상기 정보로 추출하는 것임을 특징으로 하는 상기 방법.

【청구항 3】

제1항에 있어서,

상기 제1버전 IP 어드레스가 삽입된 형태의 제2버전 IP 어드레스는 제1버전 IP 호환 제2버전 IP 어드레스 혹은 제1버전 IP 매핑 제2버전 IP 어드레스임을 특징으로 하는 상기 방법.

【청구항 4】

제3항에 있어서,

상기 제1버전 IP 호환 제2버전 IP 어드레스는 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크들간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 5】

제3항에 있어서,

제1버전 IP 매핑 제2버전 IP 어드레스는 상기 제1버전 IP만을 지원하는 네트워크와, 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 6】

제1항에 있어서,

제1버전은 버전 4이며, 제2버전은 버전 6임을 특징으로 하는 상기 방법.

【청구항 7】

제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서,

TFT 를 수신하면, 상기 TFT 정보가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하는 과정과,

상기 추출한 제1버전 IP 어드레스의 제1비트들을 새로운 TFT 정보로 생성하는 과정과,

이후 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하는 과정과,

상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 8】

제7항에 있어서.

상기 제1버전 IP 어드레스가 삽입된 형태의 제2버전 IP 어드레스는 제1버전 IP 호환 제2버전 IP 어드레스 혹은 제1버전 IP 매핑 제2버전 IP 어드레스임을 특징으로 하는 상기 방법.

【청구항 9】

제8항에 있어서,

상기 제1버전 IP 호환 제2버전 IP 어드레스는 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크들간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 10】

제8항에 있어서,

제1버전 IP 매핑 제2버전 IP 어드레스는 상기 제1버전 IP만을 지원하는 네트워크와, 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 11】

제7항에 있어서,

제1버전은 버전 4이며, 제2버전은 버전 6임을 특징으로 하는 상기 방법.

【청구항 12】

제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 방법에 있어서,

사용자 단말기는 소스 IP 어드레스가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하는 과정과,

상기 사용자 단말기는 추출한 제1버전 IP 어드레스의 제1비트들을 패킷 필터 컨텐츠로 생성하고, 상기 패킷 필터 컨텐츠를 포함하는 TFT 정보를 생성하여 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 과정과,

상기 GGSN은 상기 TFT 정보를 저장하고, 이후 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하는 과정과,

상기 GGSN은 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 13】

제12항에 있어서.

상기 제1버전 IP 어드레스가 삽입된 형태의 제2버전 IP 어드레스는 제1버전 IP 호환 제2버전 IP 어드레스 혹은 제1버전 IP 매핑 제2버전 IP 어드레스임을 특징으로 하는 상기 방법.

【청구항 14】

제13항에 있어서,

상기 제1버전 IP 호환 제2버전 IP 어드레스는 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크들간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 15】

제13항에 있어서,

제1버전 IP 매핑 제2버전 IP 어드레스는 상기 제1버전 IP만을 지원하는 네트워크와, 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크간에 사용되는 어드레스임을 특징으로 하는 상기 방법.

【청구항 16】

제12항에 있어서,

제1버전은 버전 4이며, 제2버전은 버전 6임을 특징으로 하는 상기 방법.

【청구항 17】

제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 장치에 있어서,

TFT 를 수신하고, 상기 수신된 TFT 정보가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하여 새로운 TFT 정보를 생성하도록 제어하는 제어기와,

상기 수신된 TFT 정보를 상기 새로운 TFT 정보로 저장하는 메모리를 포함함을 특징으로 하는 상기 장치.

【청구항 18】

제17항에 있어서,

상기 제어기는 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하고, 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 TFT 패킷 필터링 프로시저를 포함함을 특징으로 하는 상기 장치.

【청구항 19】

제17항에 있어서.

상기 제1버전 IP 어드레스가 삽입된 형태의 제2버전 IP 어드레스는 제1버전 IP 호환 제2버전 IP 어드레스 혹은 제1버전 IP 매핑 제2버전 IP 어드레스임을 특징으로 하는 상기 장치.

【청구항 20】

제19항에 있어서,

상기 제1버전 IP 호환 제2버전 IP 어드레스는 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크들간에 사용되는 어드레스임을 특징으로 하는 상기 장치.

【청구항 21】

제19항에 있어서,

제1버전 IP 매핑 제2버전 IP 어드레스는 상기 제1버전 IP만을 지원하는 네트워크와, 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크간에 사용되는 어드레스임을 특징으로 하는 상기 장치.

【청구항 22】

제18항에 있어서,

제1버전은 버전 4이며, 제2버전은 버전 6임을 특징으로 하는 상기 장치.

【청구항 23】

제1비트들로 구성된 제1 버전 인터넷 프로토콜(IP: Internet Protocol) 어드레스와 상기 제1비트들을 포함하는 제2비트들로 구성된 제2 버전 IP 어드레스를 지원하는 이동 통신 시스템에서 IP 버전에 따른 트래픽 플로우 탬플릿(TFT: Traffic Flow Template) 패킷 필터링 장치에 있어서,

소스 IP 어드레스가 상기 제1 버전 IP 어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우, 상기 제2 버전 IP 어드레스에 포함되어 있는 상기 제1버전 IP 어드레스의 제1비트들을 추출하여 TFT 정보를 생성하고, 상기 생성한 TFT 정보를 게이트웨이 패킷 무선 서비스 지원 노드(GGSN: Gateway GPRS Support Node)로 전송하는 사용자 단말기와,

상기 사용자 단말기로부터 수신한 TFT 정보를 저장하고, 이후 수신되는 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하고, 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 GGSN을 포함함을 특징으로 하는 상기 장치.

【청구항 24】

제23항에 있어서,

상기 GGSN은;

상기 수신 패킷 데이터의 IP 어드레스의 버전이 제2버전이고, 그 형태가 상기 제1 버전 IP어드레스가 삽입된 형태의 제2 버전 IP 어드레스일 경우 그 제2 버전 IP 어드레스에 포함되어 있는, 제1 버전 IP 어드레스를 나타내는 제1비트들을 추출하고, 상기 수신 패킷 데이터에서 추출한 제1비트들을 가지고 TFT 필터링하는 TFT 패킷 필터링 프로시저와,

상기 사용자 단말기로부터 수신된 TFT 정보를 저장하는 메모리를 포함함을 특징으로 하는 상기 장치.

【청구항 25】

제23항에 있어서.

상기 제1버전 IP 어드레스가 삽입된 형태의 제2버전 IP 어드레스는 제1버전 IP 호환 제2버전 IP 어드레스 혹은 제1버전 IP 매핑 제2버전 IP 어드레스임을 특징으로 하는 상기 장치.

【청구항 26】

제25항에 있어서,

상기 제1버전 IP 호환 제2버전 IP 어드레스는 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크들간에 사용되는 어드레스임을 특징으로 하는 상기 장치.

【청구항 27】

제25항에 있어서,

제1버전 IP 매핑 제2버전 IP 어드레스는 상기 제1버전 IP만을 지원하는 네트워크와, 상기 제1버전 IP 및 제2버전 IP 모두를 지원 가능한 네트워크간에 사용되는 어드레스임을 특징으로 하는 상기 장치.

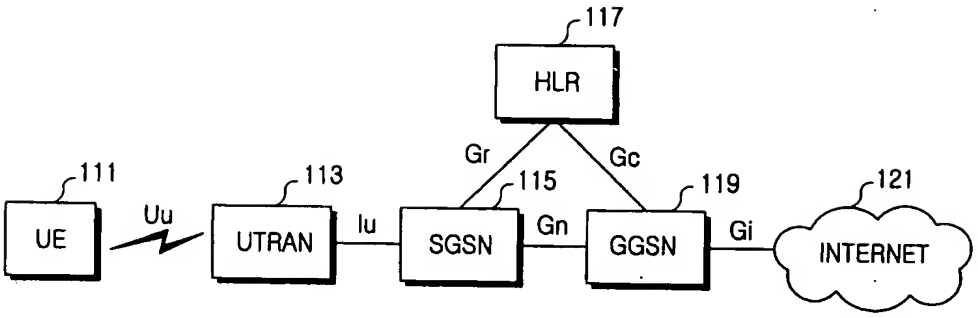
【청구항 28】

제23항에 있어서,

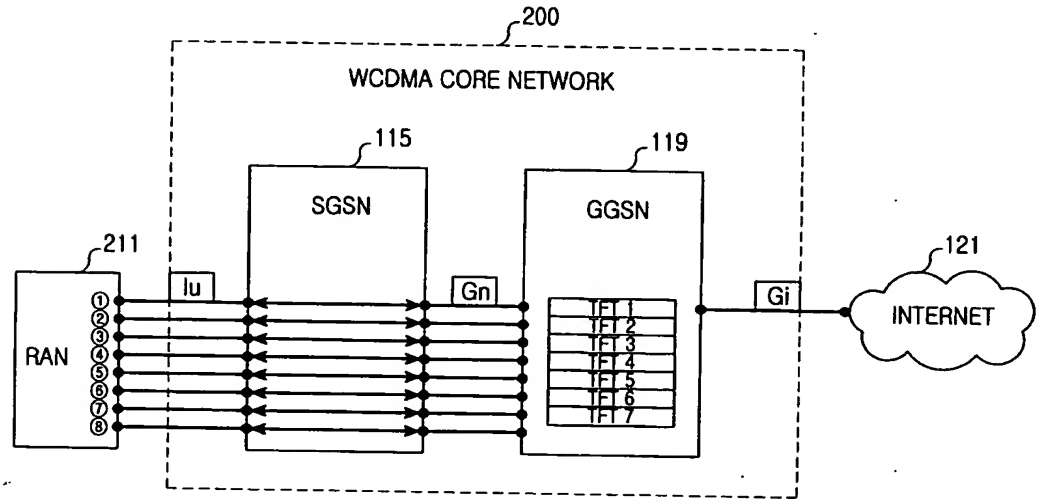
제1버전은 버전 4이며, 제2버전은 버전 6임을 특징으로 하는 상기 장치.

【도면】

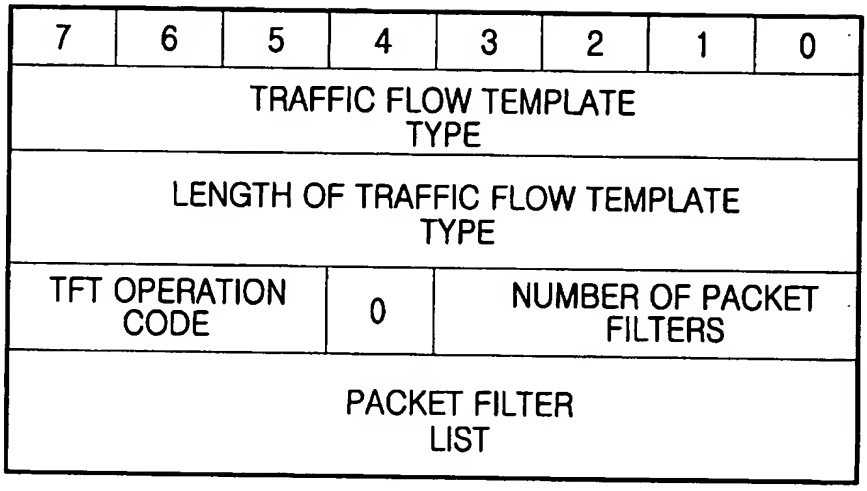
【도 1】


117
HLR
Gr
Gc
111
113
115
119
121
UE
Uu
UTRAN
Iu
SGSN
Gn
GGSN
Gi
INTERNET

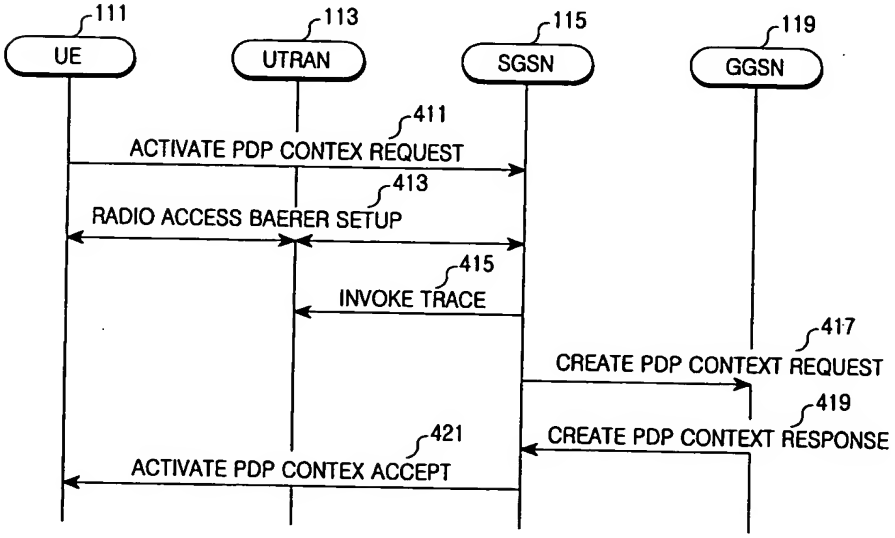
【도 2】


200
WCDMA CORE NETWORK
115
119
SGSN
GGSN
211
121
Iu
Gn
Gi
INTERNET
RAN
TFT 1
TFT 2
TFT 3
TFT 4
TFT 5
TFT 6
TFT 7

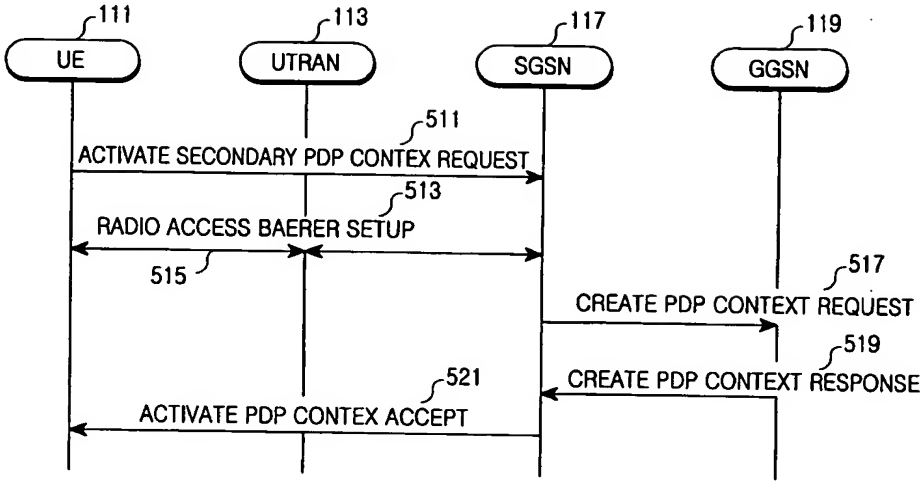
【도 3】


7
6
5
4
3
2
1
0
TRAFFIC FLOW TEMPLATE TYPE
LENGTH OF TRAFFIC FLOW TEMPLATE TYPE
TFT OPERATION CODE
0
NUMBER OF PACKET FILTERS
PACKET FILTER LIST

【도 4】


111
UE
113
UTRAN
115
SGSN
119
GGSN
411
ACTIVATE PDP CONTEX REQUEST
413
RADIO ACCESS BAERER SETUP
415
INVOKE TRACE
417
CREATE PDP CONTEXT REQUEST
419
CREATE PDP CONTEXT RESPONSE
421
ACTIVATE PDP CONTEX ACCEPT

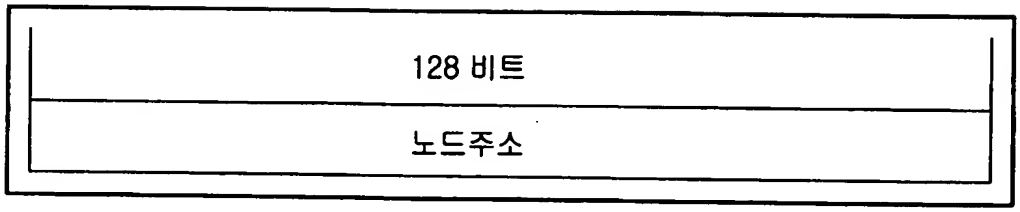
【도 5】



【도 6】

7	6	5	4	3	2	1	0
TRAFFIC FLOW TEMPLATE TYPE							
LENGTH OF TRAFFIC FLOW TEMPLATE TYPE							
TFT OPERATION CODE			0	NUMBER OF PACKET FILTERS			
PACKET FILTER IDENTIFIER 1							
PACKET FILTER EVALUATION PRECEDENCE 1							
LENGTH OF PACKET FILTER CONTENTS 1							
PACKET FILTER CONTENTS 1							
PACKET FILTER IDENTIFIER 2							
PACKET FILTER EVALUATION PRECEDENCE 2							
LENGTH OF PACKET FILTER CONTENTS 2							
PACKET FILTER CONTENTS 2							
. . . . .							

【도 7】



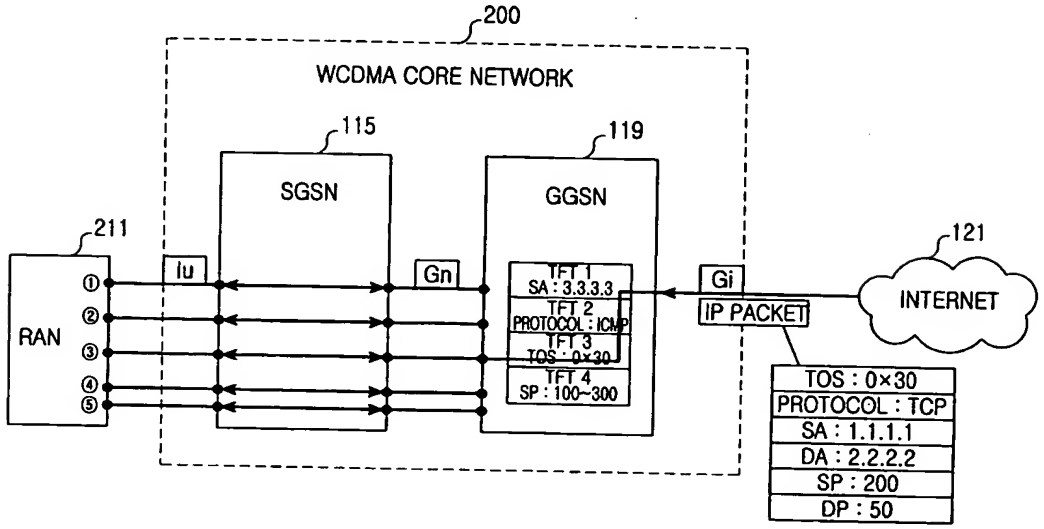
【도 8】

7	6	5	4	3	2	1	0
TRAFFIC FLOW TEMPLATE TYPE							
LENGTH OF TRAFFIC FLOW TEMPLATE TYPE							
TFT OPERATION CODE			0	NUMBER OF PACKET FILTERS			

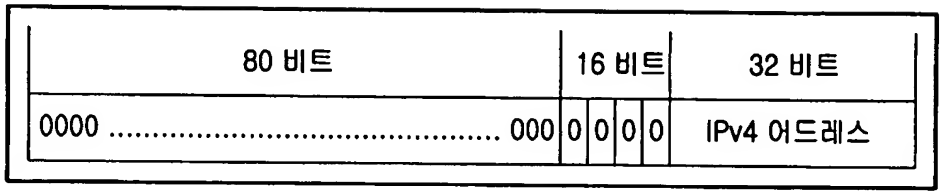
【도 9】

7	6	5	4	3	2	1	0
TRAFFIC FLOW TEMPLATE TYPE							
LENGTH OF TRAFFIC FLOW TEMPLATE TYPE							
TFT OPERATION CODE			0	NUMBER OF PACKET FILTERS			
PACKET FILTER IDENTIFIER 1							
PACKET FILTER IDENTIFIER 2							
. . . . .							
PACKET FILTER IDENTIFIER N							

【도 10】

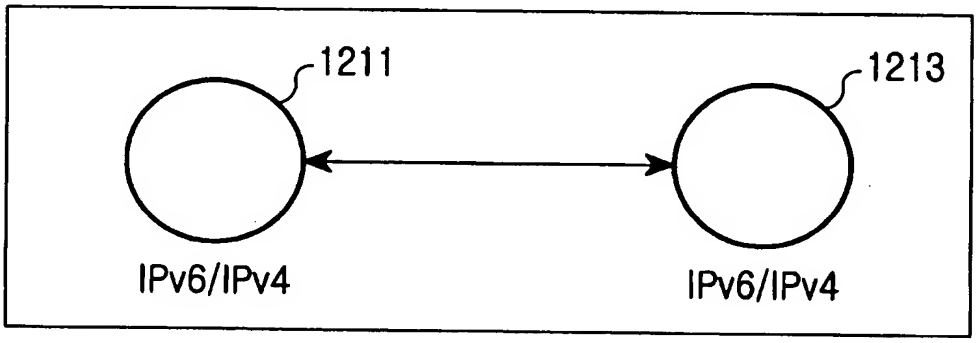


【도 11】

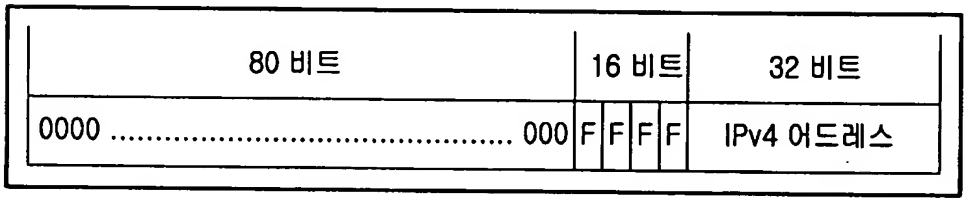


80 비트	16 비트				32 비트
0000 000	0	0	0	0	IPv4 어드레스

【도 12】

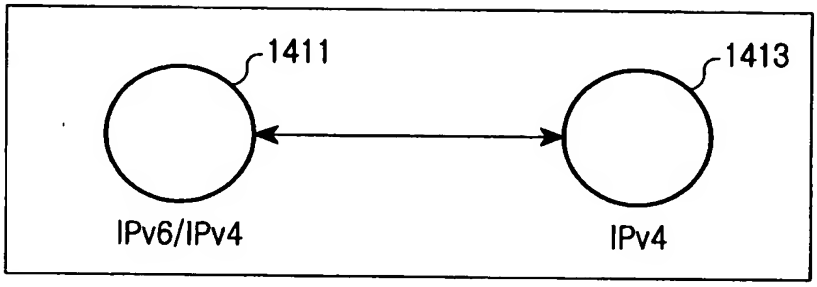


【도 13】

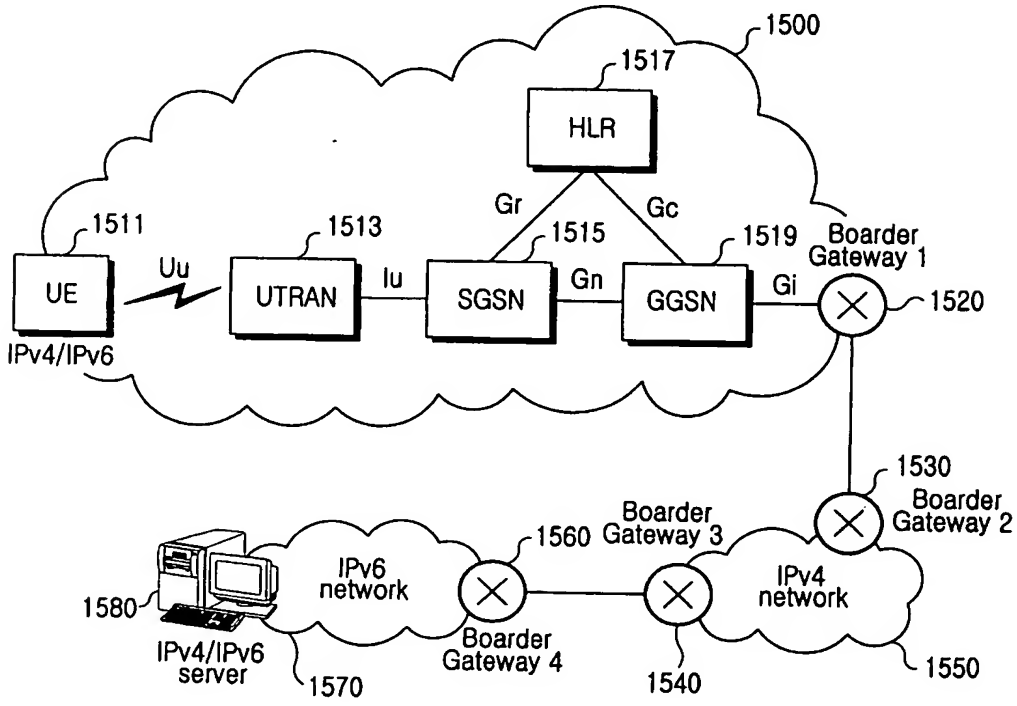


80 비트	16 비트				32 비트
0000 000	F	F	F	F	IPv4 어드레스

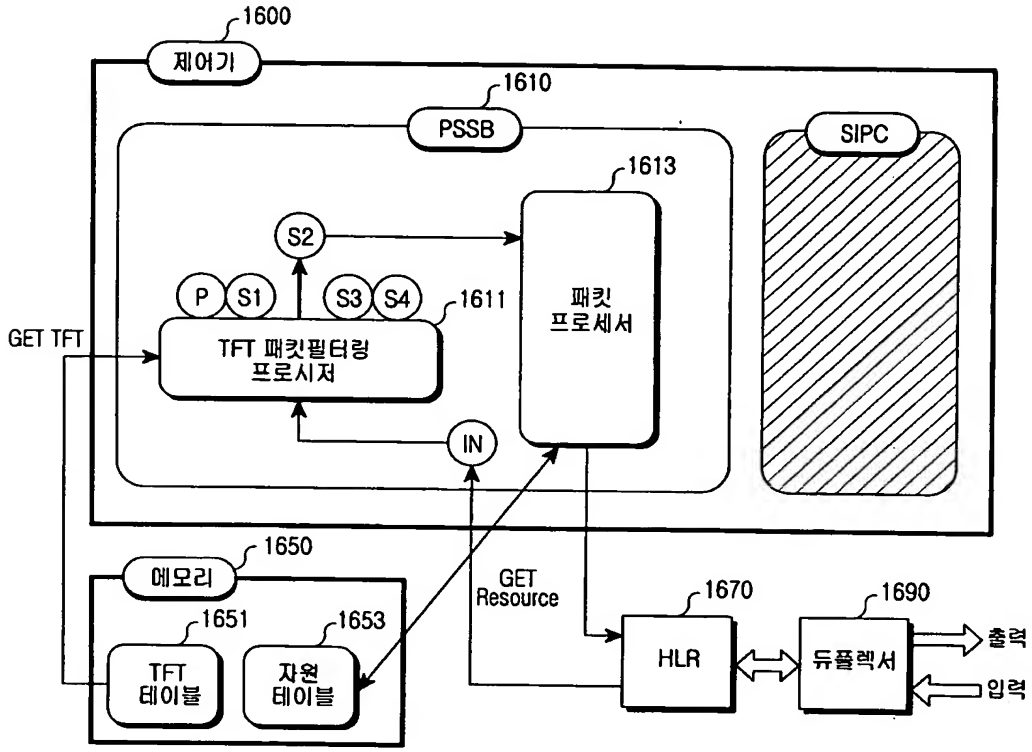
【도 14】


1411
1413
IPv6/IPv4
IPv4

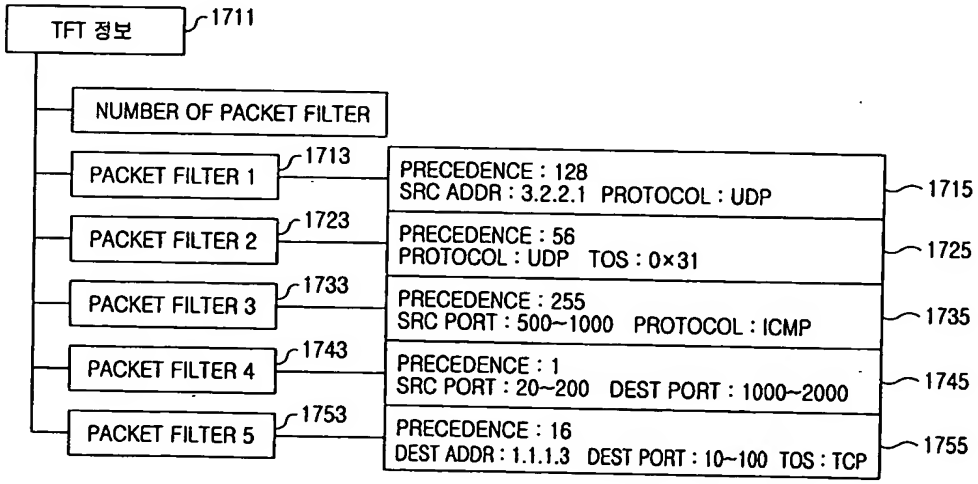
【도 15】


1500
1517
HLR
Gr
Gc
1511
1513
1515
1519
Boarder Gateway 1
UE
Uu
UTRAN
Iu
SGSN
Gn
GGSN
Gi
1520
IPv4/IPv6
1530
Boarder Gateway 2
Boarder Gateway 3
1560
IPv6 network
IPv4 network
1580
IPv4/IPv6 server
Boarder Gateway 4
1570
1540
1550

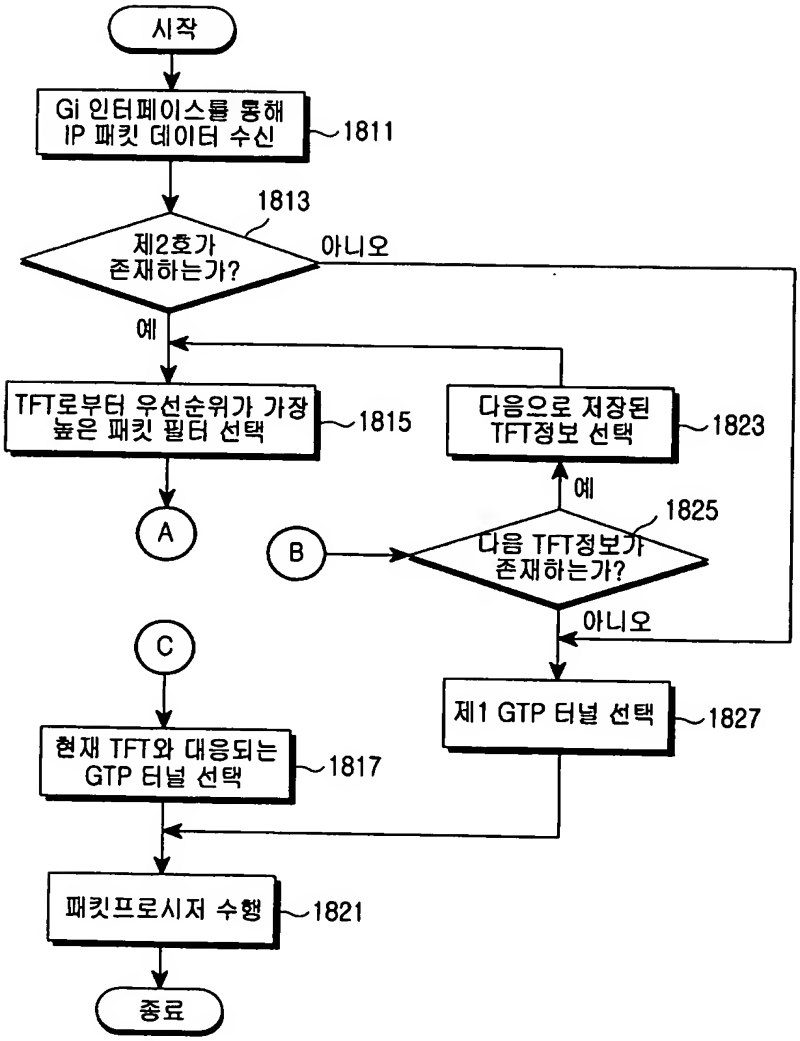
【도 16】


1600
제어기
1610
PSSB
SIPC
1613
S2
P
S1
S3
S4
1611
패킷
프로세서
GET TFT
TFT 패킷필터링
프로시저
IN
1650
메모리
GET
Resource
1670
1690
1651
1653
TFT
테이블
자원
테이블
HLR
듀플렉서
출력
입력

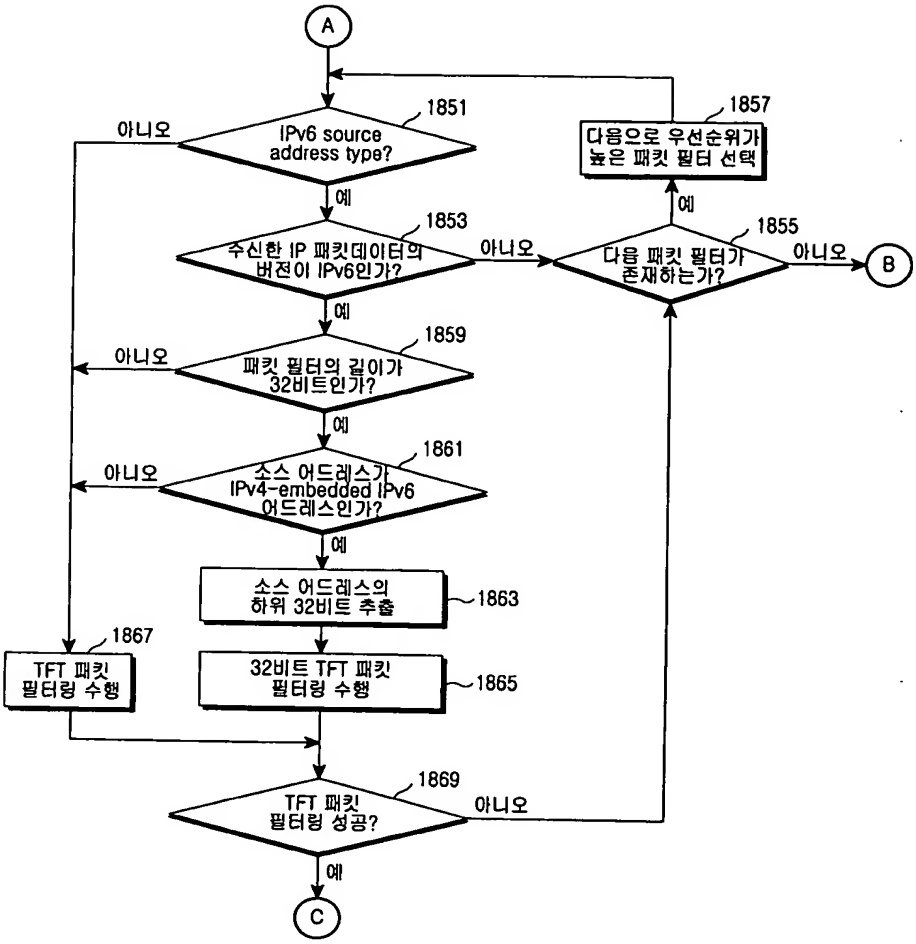
【도 17】


TFT 정보
1711
NUMBER OF PACKET FILTER
PACKET FILTER 1
1713
PRECEDENCE : 128
SRC ADDR : 3.2.2.1 PROTOCOL : UDP
1715
PACKET FILTER 2
1723
PRECEDENCE : 56
PROTOCOL : UDP TOS : 0×31
1725
PACKET FILTER 3
1733
PRECEDENCE : 255
SRC PORT : 500~1000 PROTOCOL : ICMP
1735
PACKET FILTER 4
1743
PRECEDENCE : 1
SRC PORT : 20~200 DEST PORT : 1000~2000
1745
PACKET FILTER 5
1753
PRECEDENCE : 16
DEST ADDR : 1.1.1.3 DEST PORT : 10~100 TOS : TCP
1755

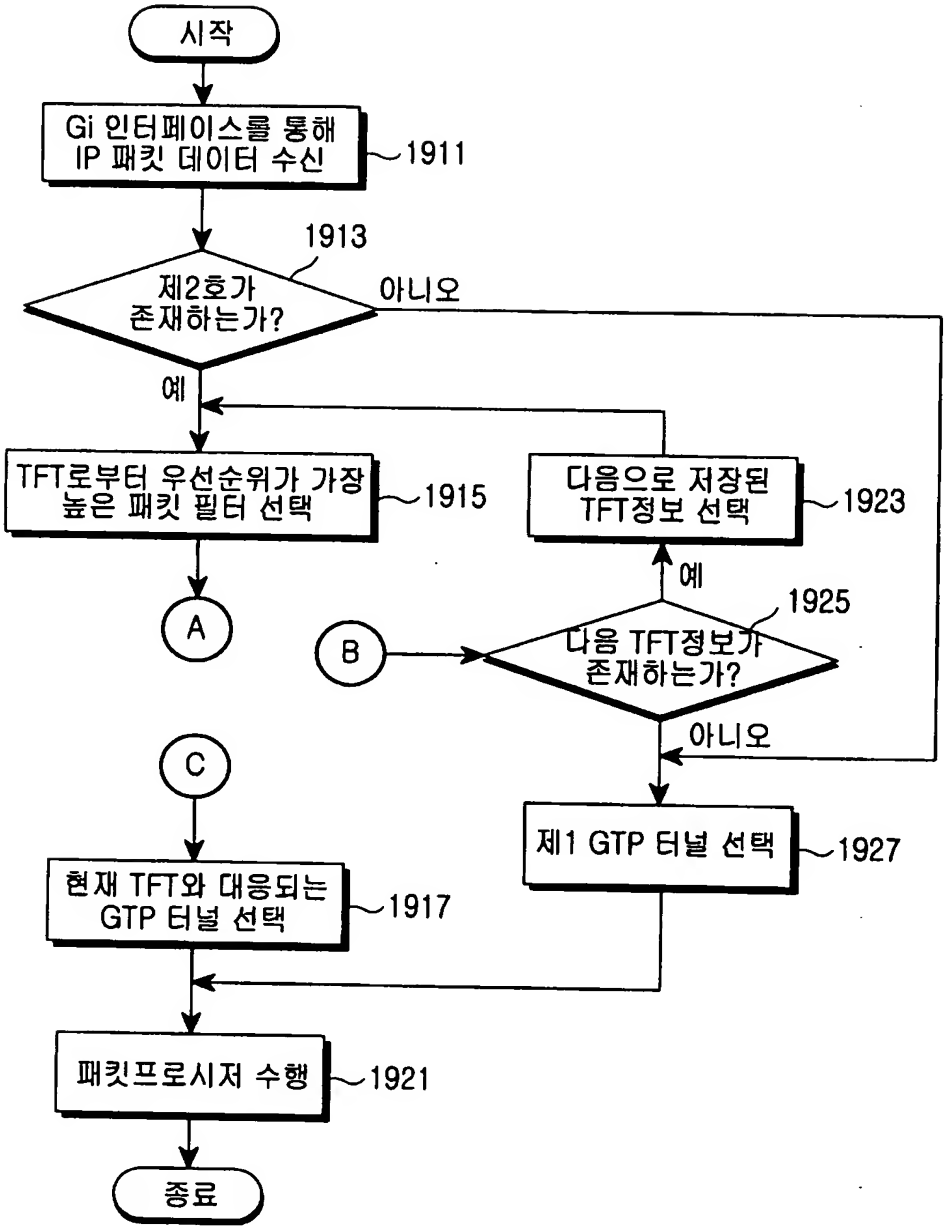
【도 18a】


시작
Gi 인터페이스를 통해 IP 패킷 데이터 수신
1811
1813
제2호가 존재하는가?
아니오
예
TFT로부터 우선순위가 가장 높은 패킷 필터 선택
1815
다음으로 저장된 TFT정보 선택
1823
A
B
예
1825
다음 TFT정보가 존재하는가?
아니오
C
제1 GTP 터널 선택
1827
현재 TFT와 대응되는 GTP 터널 선택
1817
패킷프로시저 수행
1821
종료

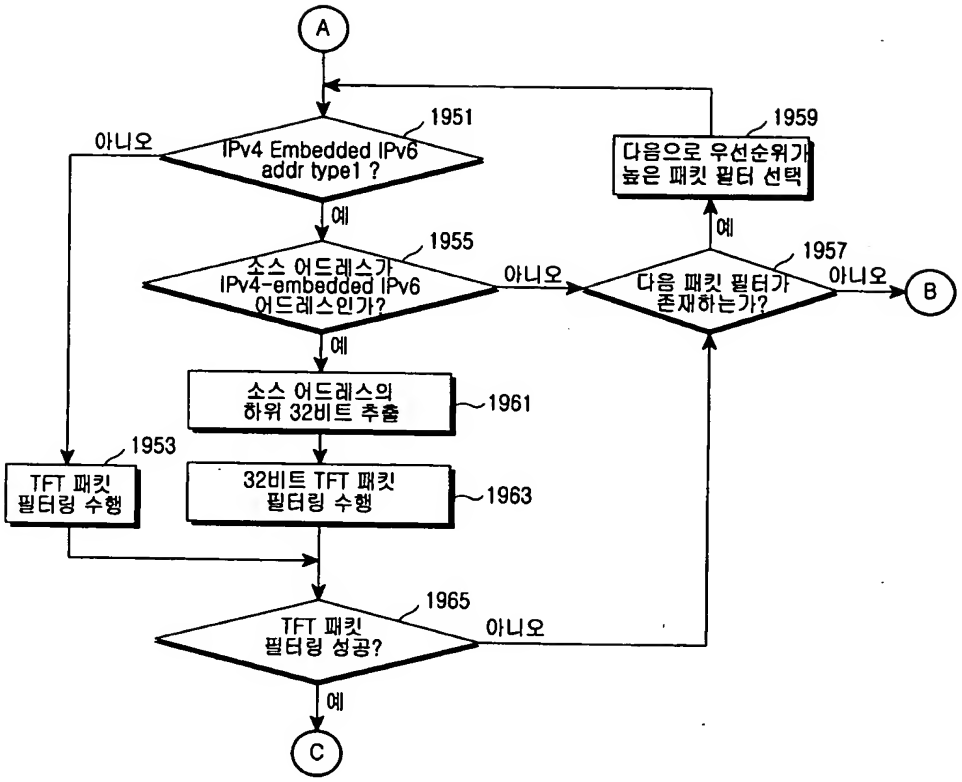
【도 18b】


A
1851
IPv6 source address type?
아니오
예
1853
수신한 IP 패킷데이터의 버전이 IPv6인가?
아니오
예
1855
다음 패킷 필터가 존재하는가?
아니오
B
예
1857
다음으로 우선순위가 높은 패킷 필터 선택
1859
패킷 필터의 길이가 32비트인가?
아니오
예
1861
소스 어드레스가 IPv4-embedded IPv6 어드레스인가?
아니오
예
소스 어드레스의 하위 32비트 추출
1863
32비트 TFT 패킷 필터링 수행
1865
TFT 패킷 필터링 수행
1867
1869
TFT 패킷 필터링 성공?
아니오
예
C

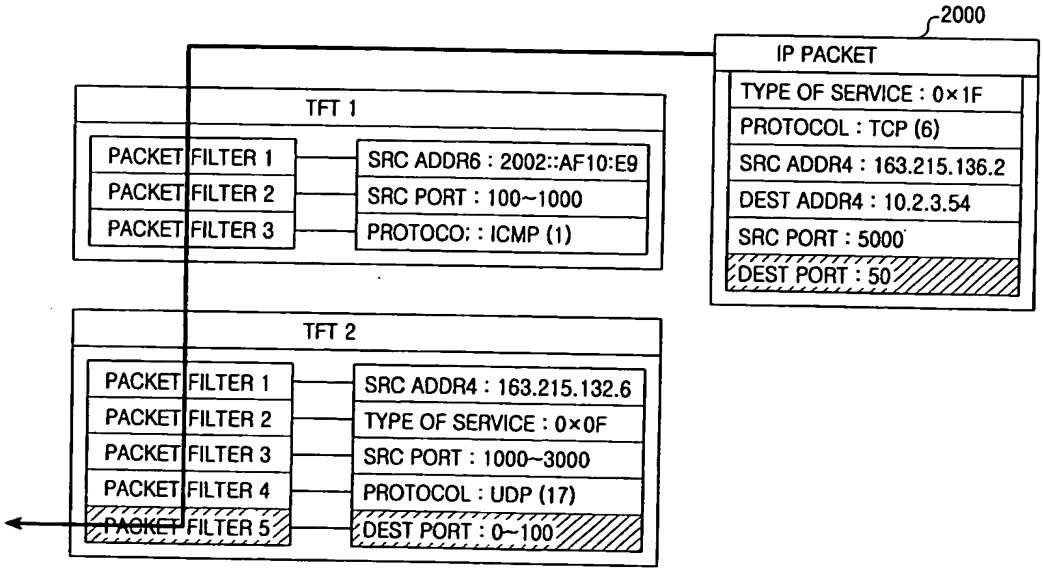
【도 19a】


시작
Gi 인터페이스를 통해 IP 패킷 데이터 수신
1911
1913
제2호가 존재하는가?
아니오
예
TFT로부터 우선순위가 가장 높은 패킷 필터 선택
1915
다음으로 저장된 TFT정보 선택
1923
A
B
예
1925
다음 TFT정보가 존재하는가?
아니오
C
제1 GTP 터널 선택
1927
현재 TFT와 대응되는 GTP 터널 선택
1917
패킷프로시저 수행
1921
종료

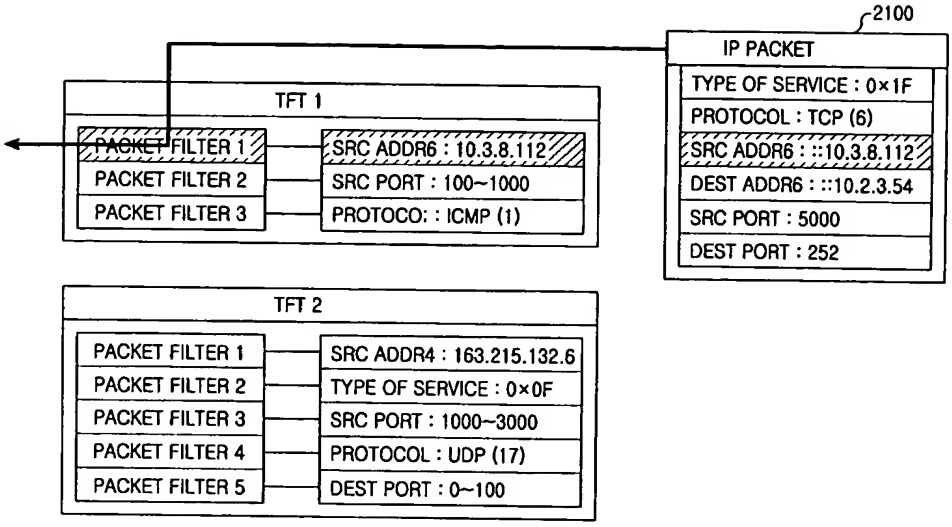
【도 19b】


A
1951
IPv4 Embedded IPv6 addr type1 ?
아니오
예
1955
소스 어드레스가 IPv4-embedded IPv6 어드레스인가?
아니오
예
소스 어드레스의 하위 32비트 추출
1961
1953
TFT 패킷 필터링 수행
32비트 TFT 패킷 필터링 수행
1963
1965
TFT 패킷 필터링 성공?
아니오
예
C
1959
다음으로 우선순위가 높은 패킷 필터 선택
예
1957
다음 패킷 필터가 존재하는가?
아니오
B

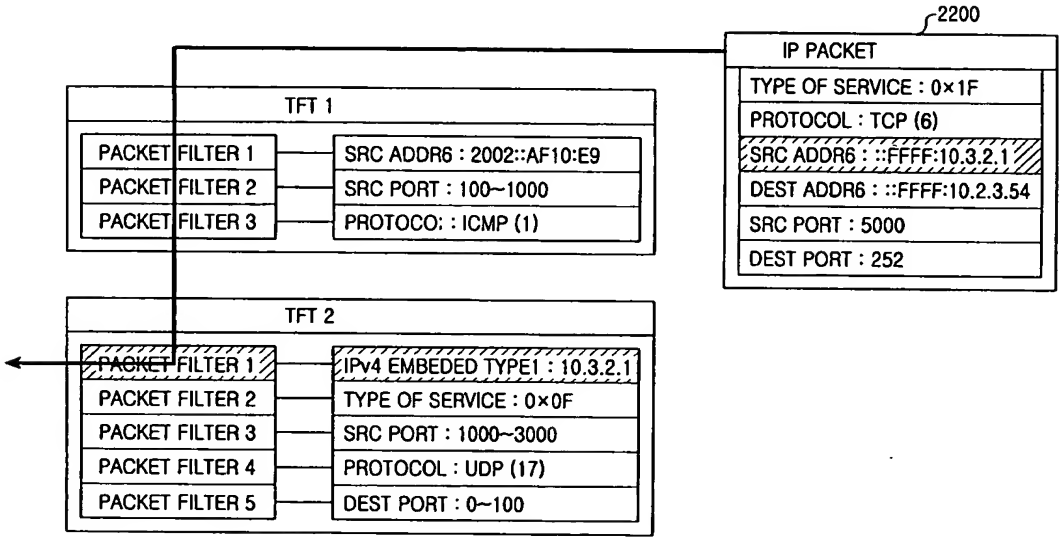
【도 20】


2000
IP PACKET
TYPE OF SERVICE : 0×1F
PROTOCOL : TCP (6)
SRC ADDR4 : 163.215.136.2
DEST ADDR4 : 10.2.3.54
SRC PORT : 5000
DEST PORT : 50
TFT 1
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
SRC ADDR6 : 2002::AF10:E9
SRC PORT : 100~1000
PROTOCOL : ICMP (1)
TFT 2
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
PACKET FILTER 4
PACKET FILTER 5
SRC ADDR4 : 163.215.132.6
TYPE OF SERVICE : 0×0F
SRC PORT : 1000~3000
PROTOCOL : UDP (17)
DEST PORT : 0~100

【도 21】


2100
IP PACKET
TYPE OF SERVICE : 0×1F
PROTOCOL : TCP (6)
SRC ADDR6 : ::10.3.8.112
DEST ADDR6 : ::10.2.3.54
SRC PORT : 5000
DEST PORT : 252
TFT 1
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
SRC ADDR6 : 10.3.8.112
SRC PORT : 100~1000
PROTOCO: : ICMP (1)
TFT 2
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
PACKET FILTER 4
PACKET FILTER 5
SRC ADDR4 : 163.215.132.6
TYPE OF SERVICE : 0×0F
SRC PORT : 1000~3000
PROTOCOL : UDP (17)
DEST PORT : 0~100

【도 22】


2200
IP PACKET
TYPE OF SERVICE : 0×1F
PROTOCOL : TCP (6)
SRC ADDR6 : ::FFFF:10.3.2.1
DEST ADDR6 : ::FFFF:10.2.3.54
SRC PORT : 5000
DEST PORT : 252
TFT 1
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
SRC ADDR6 : 2002::AF10:E9
SRC PORT : 100~1000
PROTOCO: : ICMP (1)
TFT 2
PACKET FILTER 1
PACKET FILTER 2
PACKET FILTER 3
PACKET FILTER 4
PACKET FILTER 5
IPv4 EMBEDED TYPE1 : 10.3.2.1
TYPE OF SERVICE : 0×0F
SRC PORT : 1000~3000
PROTOCOL : UDP (17)
DEST PORT : 0~100

【도 23】

필터링 회수	128 비트 연산량		32 비트 연산량		성능 개선 비트 연산량
	주소	마스크	주소	마스크	
1,000	128,000	128,000	32,000	32,000	192,000
10,000	1,280,000	1,280,000	320,000	320,000	1,920,000
100,000	12,800,000	12,800,000	3,200,000	3,200,000	19,200,000
1,000,000	128,000,000	128,000,000	32,000,000	32,000,000	192,000,000

【도 24】